



# TECNOLOGÍA IBEACON APLICADA A LA MONITORIZACIÓN DE CLIENTES

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA INFORMÁTICA

AUTOR: ADRIÁN MOLINIER GÓMEZ  
TUTOR: JESÚS M. VEGAS HERNÁNDEZ



## AGRADECIMIENTOS

En primer lugar, quiero mostrar mi agradecimiento a mis padres, a mi hermano y mi novia, que han estado apoyándome durante mi andadura en esta universidad, aconsejándome y ayudándome en los momentos difíciles que han surgido durante mi carrera.

En segundo lugar, a mis compañeros y amigos de la universidad, por todos esos momentos de alegrías y penas que he vivido junto a ellos. Sin ellos nada de esto habría sido posible.

A aquellos compañeros de clase con los que he tenido el placer de colaborar durante las prácticas y las asignaturas de la carrera, gracias.

Al resto de mis amigos, los que me soportan cada fin de semana, gracias por compartir conmigo los momentos de éxito y por estar ahí apoyándome cuando lo he pasado mal.

Y por último, pero no menos importante, agradecer enormemente su labor a mi tutor, Jesús, por responder a mis dudas y por guiarme a lo largo de este trabajo.

Muchas gracias a todos.



## RESUMEN

En el presente trabajo se desarrolla una aplicación para la monitorización de usuarios a través de la tecnología iBeacon de Apple. Este documento refleja las decisiones de diseño, funcionalidades y validaciones realizadas, tanto en el apartado de la aplicación móvil, como en el de la aplicación web, aportando finalmente una serie de valoraciones acerca de los resultados obtenidos y sobre el desarrollo del trabajo en general.



## TABLA DE CONTENIDO

Agradecimientos.....	2
Resumen.....	3
Índice de ilustraciones.....	7
Índice de tablas.....	8
Introducción.....	11
Motivación .....	11
Objetivos .....	13
Estructura del documento.....	14
Glosario de términos empleados.....	15
Estado del arte.....	17
Introducción Histórica a la Tecnología Bluetooth .....	17
Descripción de BLE y sus diferencias con Bluetooth.....	18
Descripción tecnología iBeacon .....	19
¿Qué es iBeacon? .....	19
Estudio de alternativas.....	21
WIFI.....	21
NFC .....	22
RFID .....	23
Hardware y apis de fabricantes.....	24
Limitaciones de Ibeacon.....	26
Seguridad.....	26
Precisión .....	26
Detección de dispositivos .....	27
Casos de utilización.....	29
Ejemplos de Implantación.....	30
Mcdonald's.....	30
Jivr.....	30
Aena.....	31



MLB.....	31
Fitur.....	31
Smarth Costa del Sol.....	31
Descripción de la solución propuesta .....	33
Análisis de la propuesta .....	35
Definición de casos de uso .....	35
Requisitos del sistema.....	42
Requisitos funcionales.....	43
Requisitos no funcionales .....	51
Diseño e implementación de la propuesta .....	53
Diseño lógico .....	53
Diagrama de Paquetes .....	54
Diagrama de Despliegue .....	55
Diagrama de Modelo de los datos .....	56
Diseño físico.....	57
Fases de Implementación.....	59
Aplicación Móvil .....	59
Aplicación Web .....	62
Conexión Back/Front End .....	68
Validación y evaluación.....	69
Validación de requisitos .....	69
Gestión del proyecto.....	77
Planificación inicial.....	77
Planificación final .....	78
Presupuesto .....	79
Conclusiones.....	81
Comentarios finales .....	81
Líneas futuras.....	81
Referencias .....	83
Anexo I: Guía rápida de configuración de un iBeacon .....	85



Anexo II: Diagrama de planificación .....	89
---	----



## ÍNDICE DE ILUSTRACIONES

<i>Ilustración 1: Conjunto de iBeacons de Estimote [7]</i> .....	19
<i>Ilustración 2: Pago mediante NFC[11]</i> .....	22
<i>Ilustración 3: Escaneo tarjeta RFID [12]</i> .....	23
<i>Ilustración 4: Detección iBeacon [15]</i> .....	28
<i>Ilustración 5: Diseño Jivr [17]</i> .....	30
<i>Ilustración 6: Gráfico arquitectura sistema</i> .....	34
<i>Ilustración 7: Diagrama casos de uso</i> .....	35
<i>Ilustración 8: Diagrama de paquetes</i> .....	54
<i>Ilustración 9: Diagrama de despliegue</i> .....	55
<i>Ilustración 10: Diagrama modelo de datos</i> .....	56
<i>Ilustración 11: Pantalla inicio aplicación móvil</i> .....	62
<i>Ilustración 12: Pantalla login aplicación web</i> .....	63
<i>Ilustración 13: Pantalla registro aplicación web</i> .....	64
<i>Ilustración 14: Pantalla iBeacons aplicación web</i> .....	65
<i>Ilustración 15: Pantalla estadísticas aplicación web</i> .....	66
<i>Ilustración 16: Menú aplicación web</i> .....	67
<i>Ilustración 17: Menú adaptación móvil aplicación web</i> .....	67
<i>Ilustración 18: Menú adaptación móvil desplegado aplicación web</i> .....	67
<i>Ilustración 19: Aplicación Estimote en App Store</i> .....	85
<i>Ilustración 20: Menú principal aplicación Estimote</i> .....	86
<i>Ilustración 21: Sección radar aplicación Estimote</i> .....	86
<i>Ilustración 22: Sección List aplicación Estimote</i> .....	87
<i>Ilustración 23: Propiedades de un beacon detectado</i> .....	87
<i>Ilustración 24: Modificación Major aplicación Estimote</i> .....	88
<i>Ilustración 25: Diagrama de planificación inicial estimada</i> .....	89



## ÍNDICE DE TABLAS

Tabla 1: Comparativa BT/BLE[5] .....	18
Tabla 2: Ejemplo de distribución valores beacon <sup>[8]</sup> .....	20
Tabla 3: Plantilla de casos de uso .....	36
Tabla 4: CU-01 - Inicio de sesión.....	36
Tabla 5: CU-02 – Registro de usuario.....	37
Tabla 6: CU-04–Visualización estadísticas.....	37
Tabla 7: CU-05 – Visualizar iBeacons .....	37
Tabla 8: CU-06–Dar de alta iBeacon.....	38
Tabla 9: CU-07 – Dar de baja iBeacon.....	38
Tabla 10: CU-08–Modificar iBeacon .....	38
Tabla 11: CU-09–Visualizar estudios.....	38
Tabla 12: CU-10 – Dar de alta estudio.....	39
Tabla 13: CU-11 –Dar de baja estudio .....	39
Tabla 14: CU-12–Modificar estudio .....	39
Tabla 15: CU-13–Visualizar usuarios .....	40
Tabla 16: CU-14–Dar de baja usuario.....	40
Tabla 17: CU-15–Modificar usuario .....	40
Tabla 18: CU-16–Exportar a excel.....	40
Tabla 19: CU-17–Exportar a csv .....	41
Tabla 20: Plantilla de requisitos.....	42
Tabla 21: RF-app-001 .....	43
Tabla 22: RF-app-002.....	43
Tabla 23: RF-app-003 .....	44
Tabla 24: RF-app-004.....	44
Tabla 25: RF-app-005 .....	44
Tabla 26: RF-serv-001 .....	45
Tabla 27: RF-serv-002.....	45
Tabla 28: RF-serv-003 .....	45





Tabla 29: RF-serv-004 .....	45
Tabla 30: RF-serv-005.....	46
Tabla 31: RF-serv-006.....	46
Tabla 32: RF-serv-007.....	46
Tabla 33: RF-serv-008.....	47
Tabla 34: RF-serv-009 .....	47
Tabla 35: RF-serv-010.....	47
Tabla 36: RF-serv-011.....	47
Tabla 37: RF-serv-012 .....	48
Tabla 38: RF-serv-013.....	48
Tabla 39: RF-serv-014.....	48
Tabla 40: RF-serv-015.....	49
Tabla 41: RF-serv-016.....	49
Tabla 42: RF-serv-017.....	49
Tabla 43: RF-serv-018.....	49
Tabla 44: RF-serv-019 .....	50
Tabla 45: RNF-app-001 .....	51
Tabla 46: RNF-app-002 .....	51
Tabla 47: RNF-app-003.....	51
Tabla 48: RNF-app-004 .....	52
Tabla 49: RNF-serv-001.....	52
Tabla 50: RNF-serv-002.....	52
Tabla 51: Ordenador de desarrollo .....	57
Tabla 52: Ordenador de pruebas .....	58
Tabla 53: Ordenador de pruebas .....	58
Tabla 54: Plantilla de validación de requisitos .....	69
Tabla 55: Tabla de validación para RF-app-001 .....	70
Tabla 56: Tabla de validación para RF-app-002 .....	70
Tabla 57: Tabla de validación para RF-app-003.....	70
Tabla 58: Tabla de validación para RF-app-004 .....	70



Tabla 59: Tabla de validación para RF-app-005 .....	71
Tabla 60: Tabla de validación para RF-serv-001.....	71
Tabla 61: Tabla de validación para RF-serv-002.....	71
Tabla 62: Tabla de validación para RF-serv-003.....	71
Tabla 63: Tabla de validación para RF-serv-004.....	71
Tabla 64: Tabla de validación para RF-serv-005 .....	72
Tabla 65: Tabla de validación para RF-serv-006 .....	72
Tabla 66: Tabla de validación para RF-serv-007.....	72
Tabla 67: Tabla de validación para RF-serv-008.....	72
Tabla 68: Tabla de validación para RF-serv-009 .....	73
Tabla 69: Tabla de validación para RF-serv-010.....	73
Tabla 70: Tabla de validación para RF-serv-011 .....	73
Tabla 71: Tabla de validación para RF-serv-012 .....	73
Tabla 72: Tabla de validación para RF-serv-013 .....	73
Tabla 73: Tabla de validación para RF-serv-014 .....	74
Tabla 74: Tabla de validación para RF-serv-015 .....	74
Tabla 75: Tabla de validación para RF-serv-016 .....	74
Tabla 76: Tabla de validación para RF-serv-017 .....	74
Tabla 77: Tabla de validación para RF-serv-018 .....	74
Tabla 78: Tabla de validación para RF-serv-019.....	75
Tabla 79: Tabla de validación para RNF-app-001.....	75
Tabla 80: Tabla de validación para RNF-app-002 .....	75
Tabla 81: Tabla de validación para RNF-app-003.....	75
Tabla 82: Tabla de validación para RNF-app-004 .....	75
Tabla 83: Tabla de validación para RNF-serv-001 .....	76
Tabla 84: Tabla de validación para RNF-serv-002.....	76
Tabla 85: Gastos de personal.....	79
Tabla 86: Gastos de material amortizable .....	80
Tabla 87: Presupuesto total del proyecto .....	80



## INTRODUCCIÓN

En 2013 Apple desarrolla una tecnología llamada *iBeacon*, cuya finalidad es impulsar y dar valor añadido al sector *retail*. Este sector engloba todos los comercios y superficies dedicadas a la venta física de productos al consumidor final.

En la actualidad, este sector no dispone de muchos medios para monitorizar y extraer información acerca de las preferencias y comportamientos de los usuarios en tiempo real, siendo justamente lo que esta tecnología pretende. A cambio, el usuario recibirá servicios personalizados, que pueden ayudar a crear una experiencia más individualizada y supondrán un valor añadido.

La creciente relevancia de la recolección y tratamiento masivo de datos ayudan a fomentar este tipo de tecnologías y, a pesar de ser muy novedosas, las empresas se han mostrado interesadas en investigar hasta qué punto son viables y les benefician.

## MOTIVACIÓN

Al tratarse la tecnología *Smart Bluetooth* (concretamente *iBeacon*) de un tema innovador y de gran interés, surge este trabajo de fin de grado, que intenta acercar esta tecnología a un caso práctico general, pudiéndose implantar en un futuro en diferentes escenarios.

En la actualidad, existen multitud de técnicas para la recolección de datos, mediante tarjetas de fidelización que aportan descuentos cada vez que compramos, encuestas, formularios de registro de usuarios en páginas web junto con las cookies y registros de actividad, etc...

Pero ninguna de ellas proporciona información acerca de patrones de comportamiento en tiempo real, es decir, puedes saber qué y cuándo compra habitualmente, pero no cuánto tarda en elegir entre un producto u otro, cómo recorre el establecimiento, cuánto tiempo está en cada sección de un supermercado, etc..

En el presente trabajo se pretende aportar una solución distinta a las mencionadas en el párrafo anterior, empleando la tecnología *iBeacon* (*Smart Bluetooth*) como tecnología principal, y proporcionando, por tanto, información mucho más detallada y más precisa. Para ello, se elaborará un prototipo que monitorizará a un usuario a través de su paso por los *beacons* y mandará la información a un servidor para su visualizado.

A su vez, se pretende profundizar en adquirir conocimientos sobre cómo funciona este tipo de tecnología así como todas las implicadas en el proyecto.



La motivación del autor es, principalmente, adquirir una base de conocimientos sobre la tecnología *BLE* (*Bluetooth Low Energy*), el proceso completo de desarrollo de aplicaciones *IOS*, Web y la comunicación entre ellas, así como reforzar la capacidad de esfuerzo y superación del alumno al verse inmerso en un proyecto con una larga lista de tareas y problemas por resolver.

Con este trabajo de fin de grado se espera dar el primer paso hacia proyectos y aplicaciones de mayor envergadura a lo largo de la carrera profesional del autor.



## OBJETIVOS

En el presente trabajo se pretende crear un servicio que monitorice e identifique patrones de conducta, a través de los datos enviados por una aplicación, instalada en los dispositivos de los clientes.

Objetivo general:

- Se desea como objetivo principal la recolección de datos de usuarios, monitorizando todo el comportamiento que desarrolle un cliente de un establecimiento, mediante tecnología *iBeacon*.

Objetivos secundarios:

- La monitorización debe ser lo más transparente y económica posible.
- Obtención de datos que permitan la interpretación de datos para la creación de patrones de conducta. Con los resultados obtenidos se podrán conocer gustos, fechas de entrada y salida, horas, duración de permanencia en el comercio, etc...
- Monitorización en tiempo real de los usuarios, por ejemplo, visionando en un mapa del establecimiento dónde se encuentran.



## ESTRUCTURA DEL DOCUMENTO

En este apartado se muestra la estructura que presenta el presente documento, buscando detallar la propuesta de una forma ordenada y exhaustiva.

- **Estado del arte:** Comprende una introducción histórica sobre la tecnología Bluetooth, descripción de la tecnología *iBeacon*, un estudio de alternativas a esta tecnología, sus limitaciones y algunos casos en los que ya se ha implantado este sistema.
- **Análisis de la propuesta:** Define los requisitos, funcionales y no funcionales, establecidos para el proyecto, así como los casos de uso del sistema.
- **Diseño e implementación de la propuesta:** Aporta el diseño físico-lógico de la aplicación, explicando qué y cómo se ha desarrollado de forma detallada pero intentando no entrar en demasiados tecnicismos.
- **Gestión del proyecto:** Expone la planificación estimada para este proyecto, así como el presupuesto previsto para el completo desarrollo de la aplicación.
- **Validación:** Muestra el resultado de las pruebas realizadas con la aplicación, reflejado en la validación de los requisitos determinados en el apartado de "Análisis de la propuesta".
- **Conclusiones:** Finalmente, se ofrecen una serie de conclusiones acerca del resultado final del proyecto, junto a una validación de los objetivos que se marcaron al comienzo del desarrollo, así como unas líneas futuras de cara a mejorar y perfeccionar las aplicaciones realizadas.



## GLOSARIO DE TÉRMINOS EMPLEADOS

Para una correcta comprensión del presente documento, se exponen a continuación los términos especiales y acrónimos utilizados, ordenados alfabéticamente:

- **iBeacon:** Son una balizas que funcionan con tecnología *BLE* que emiten señales de radio de baja frecuencia.
- **BLE:** *Bluetooth lowenergy*, también denominado "*Smarth Bluetooth*", es una tecnología *Bluetooth* de bajo consumo, que salió en la versión 4.0.
- **BT:** Abreviatura de *Bluetooth*.
- **SIG:** *Bluetooth Special Interest Group*. Grupo formado por todos los fabricantes de dispositivos *Bluetooth*.







## ESTADO DEL ARTE

### INTRODUCCIÓN HISTÓRICA A LA TECNOLOGÍA BLUETOOTH

A continuación se realiza una pequeña introducción a la historia y funcionamiento de la tecnología *Bluetooth*, para después centrarse en describir su vertiente de bajo consumo *BLE*.

En 1994 la compañía de telecomunicaciones Ericsson, inició una investigación cuya finalidad era eliminar cables para la comunicación entre diferentes dispositivos, mediante ondas de radio de baja potencia. Para ello, los holandeses Jaap Haartsen y Mattisson Sven [1] desarrollaron una tecnología retro compatible y de un alcance de unos diez metros.

Más tarde, tras comprobar las posibilidades de dicha tecnología, cinco compañías, Ericsson, Nokia, IBM, Toshiba e Intel, se agruparon con el objetivo de establecer una especificación global sin cables de corto alcance.

Para impulsar e incrementar las especificaciones de esta tecnología se creó el grupo *Bluetooth Special Interest Group (SIG)* [2]. Cualquier empresa que desee incorporar esta tecnología a sus dispositivos, debe formar parte de este grupo. Entre ellos los más activos son *Ericsson, Intel, Lenovo, Microsoft, Motorola, Nokia* y *Toshiba*.

Actualmente la tecnología Bluetooth está presente en multitud de dispositivos electrónicos y su uso se ha generalizado con la llegada de los *smartphones*.

Como curiosidad decir que Ericsson bautizó a esta tecnología con el nombre de *Bluetooth* debido al rey *Harald Blatand* [3]. Este rey, bautizado como rey *Bluetooth*, gobernó durante el siglo X los territorios que hoy conocemos con el nombre de Dinamarca, Noruega y Suecia. Se caracterizó por gobernar y unificar sus reinos sobre una misma religión, la cristiana.

La idea de los desarrolladores era crear un nombre que no fuese muy técnico, fácil de recordar y que se hiciese conocido.

El estándar *Bluetooth* actualmente funciona emitiendo una señal inalámbrica de tipo radio. Utiliza la frecuencia de 2,45 GHz y transmiten una señal de 1 mW[4] a una distancia de entre 10 y 100 metros.

En 2009, el grupo *SIG* creó un nuevo estándar, denominado *Bluetooth 4.0*. Con este nuevo estándar nació lo que se conoce como *Smart Bluetooth (BLE)*. La descripción de *BLE* junto con sus semejanzas y diferencias de *Bluetooth* tradicional se describen a continuación, en el siguiente apartado.



## DESCRIPCIÓN DE BLE Y SUS DIFERENCIAS CON BLUETOOTH

*Bluetooth Low Energy (BLE)* es una tecnología sin cables que permite la comunicación entre dispositivos compatibles a baja distancia. Tiene características muy parecidas al estándar *Bluetooth*, pero es más económico (entre un 60% y un 80% más barato), menor consumo energético y es perfecto para aplicaciones simples que no requieren de mucho intercambio de información. En el caso de tener aplicaciones más complejas es preferible el uso de *Bluetooth* convencional.

A continuación se muestra una comparativa entre *BT* y *BLE* mostrando sus especificaciones técnicas.

Especificación Técnica	Bluetooth tradicional	Bluetooth LowEnergy
Rango/Distancia (Máximo Teórico)	100 m	>100 m
Latencia	100 ms	6 ms
Tiempo total mínimo para enviar datos.	100 ms	3 ms
Consumo de energía	1 W	Entre 0.01 y 0.05 W
Pico máximo de consumo	<30 mA	<15 mA

Tabla 1: Comparativa BT/BLE[5]

## DESCRIPCIÓN TECNOLOGÍA IBEACON

En el año 2013 con la llegada de *iOS 7*, Apple presentó una nueva tecnología a la que bautizó con el nombre de *iBeacon*, cuyo principal objetivo era crear un valor añadido al sector de *retail*, creando por ejemplo, ofertas específicas para los clientes que se encuentren dentro de la tienda o algún tipo de fidelización.

El auge de dispositivos como los *wearables* (dispositivos para llevar puesto como gafas o relojes inteligentes) y la generalización del uso de *smartphones*, da un sinfín de nuevas posibilidades no sólo dentro del sector, sino también en una gran cantidad de situaciones diferentes.

En la actualidad, aunque esta tecnología ha sido desarrollada por *Apple*, existe compatibilidad con algunos de los dispositivos *Android*, en concreto, los modelos que dispongan de *Bluetooth 4.0*.

---

### ¿QUÉ ES IBEACON?

*Apple* define la tecnología *iBeacon* con la siguiente frase: "Una nueva clase de transmisores de bajo consumo y bajo coste que pueden notificar a dispositivos *iOS 7* de su presencia por proximidad" [6].

Se tratan de transmisores de señales *Bluetooth* de baja energía, que actúan como balizas y pueden ser detectados por dispositivos que cuenten con tecnología *Bluetooth 4.0*.



*Ilustración 1: Conjunto de iBeacons de Estimote [7]*



Las señales enviadas contienen información relativa al *beacon*, localización y nuestra proximidad, es decir, cada dispositivo que detecte la señal emitida por el *beacon* recibirá un ID y la intensidad de la señal.

Más concretamente, el ID del *beacon* se compone de los siguientes campos:

- ✓ *UUID*: Identificador único para la aplicación.
- ✓ *Major*: Ayuda a determinar un *beacon* en concreto. Por ejemplo, distinguir el *beacon* de una tienda por ciudades.
- ✓ *Minor*: Especifica otra subdivisión. Por ejemplo, en una tienda la zona de calzado.

Según la propia documentación de *Apple*, un ejemplo de la información que posee cada *beacon* sería:

Store Location		San Francisco	Paris	London
UUID		D9B9EC1F-3925-43D0-80A9-1E39D4CEA95C		
Major		1	2	3
Minor	Clothing	10	10	10
	Housewares	20	20	20
	Automotive	30	30	30

Tabla 2: Ejemplo de distribución valores beacon[8]

El alcance de las señales emitidas varía dependiendo del fabricante, pudiendo alcanzar los 70 metros, siempre y cuando no existan barreras arquitectónicas o de otra índole, que atenúen la señal.

Existen algunas funcionalidades a mayores en *beacons* de algunos fabricantes. Por ejemplo, en el caso de *Estimote*, una de las empresas más populares en este campo, los *beacons* disponen de sensores de movimiento y temperatura.

Estos sensores podrían tener como objetivo analizar patrones de comportamiento, adhiriendo un *beacon* a ciertos objetos, por ejemplo, a una bicicleta y recopilando información acerca de su actividad física o temperatura en el exterior e interior.



## ESTUDIO DE ALTERNATIVAS

A continuación, se exponen diferentes tecnologías que tienen uno o varios puntos en común con la tecnología *BLE* y se realizará un análisis comparativo mostrando las ventajas e inconvenientes de cada elección.

---

### WIFI

En este apartado se describirá la tecnología *WIFI* en su vertiente de monitorización cuyo objetivo es realizar una comparación con *BLE*, ya que por lo general ambas tecnologías son usadas en campos completamente distintos.

El rastreo por *WIFI* se realiza a través de las peticiones que realiza el *smartphone* siempre que tenga la opción *WIFI* habilitada. Estas peticiones se realizan constantemente en búsqueda de todas las posibles señales que se encuentren en el entorno del dispositivo.

Esta tecnología permite la monitorización de usuarios aunque no exista conexión activa entre el dispositivo y el emisor de señales *WIFI*. Cada vez que el *smartphone* realice una petición se guarda información del usuario, incluso sin intervención ni permisos aceptados.

Esta información consiste en la dirección *MAC* del dispositivo, el *timestamp* del momento en que se realizó la petición, el fabricante del dispositivo y la intensidad de la señal.

La diferencia básica entre esta tecnología y *BLE* reside en los modelos de privacidad, *Opt-In* y *Opt-Out*[9].

En el caso de las señales *WIFI* estamos hablando de un modelo *Opt-Out* en el que el usuario puede ser monitorizado incluso cuando no ha dado su consentimiento explícito.

En el caso de *BLE*, el usuario debe en primer lugar instalarse una aplicación en su dispositivo y activar la opción *Bluetooth* y en segundo lugar debe proporcionar su consentimiento para poder descubrir *beacons* en la región. Este último caso se trata del modelo *Opt-In*, cuya privacidad es respetada dando la posibilidad al usuario de escoger si desea ser monitorizado o no.

A la hora de la precisión ambas tecnologías tienen características muy similares a la hora de obtener el *rsi*(intensidad de la señal). *WIFI* está diseñada para precisar el punto donde se encuentra el dispositivo detectado, mientras que un *beacon* detecta diferentes casos de proximidad, lo que facilita mucho las cosas cuando queremos detectar si un usuario se ha acercado o alejado de un sitio sin importar exactamente a qué distancia se encuentra.

En cuanto a los costes de implantación varía mucho dependiendo del espacio a ser monitorizado y del fabricante. Por un lado, la tecnología *WIFI* ya se encuentra instalada de forma habitual, pero tiene un coste de energía mayor, necesitando estar conectados a la red eléctrica.

Las balizas *iBeacon* tienen un consumo mucho menor, llegando a los tres años de la duración de sus baterías, dependiendo de la configuración.

En este caso la tecnología *WIFI* para la monitorización de usuarios se ha descartado en gran medida por la falta de privacidad y que no ofrece ningún servicio a mayores para los usuarios. En el caso de *iBeacon* si el usuario es monitorizado posiblemente sea para recibir una serie de servicios personalizados y siempre siendo informado de su monitorización.

---

## NFC

*NFC* al igual que *BLE* permite la comunicación entre un emisor y un receptor sin la necesidad de cables.

En el caso de *NFC* su característica principal es el corto radio de acción que posee. Sus señales tienen un alcance de unos 20 cm como máximo. Esta particularidad hace que en ciertas situaciones sea la tecnología adecuada a implementar[10].



Ilustración 2: Pago mediante NFC[11]

*NFC* se relaciona uno a uno, es decir, sólo funciona con un dispositivo cada vez y no puede monitorizar varios simultáneamente. En el caso de *BLE*, las señales del *beacon* pueden ser recibidas por cualquier dispositivo compatible que se encuentre en el rango, siendo 1 a n desde la perspectiva de la baliza.

En cuanto a la eficiencia energética ambas son poco costosas. Por un lado, *NFC* permite incluso que uno de los dos extremos de la comunicación funcione sin necesidad de batería. Por otro lado, los dispositivos *iBeacon* funcionan con poca energía y necesitan reemplazar la batería cada cierto tiempo.

En el caso de la seguridad *BLE* sólo envía los valores que identifican a cada *beacon* por tanto no supone ningún riesgo. En el caso de *NFC* dispone de un modo seguro para las transferencias de datos sensibles como, por ejemplo, las tarjetas de crédito con *NFC* incorporado.

Todo lo dicho anteriormente indica que, para el objetivo principal de este proyecto, la monitorización es clave y *NFC* sólo lo permite en un rango muy pequeño. Por tanto, se descarta aunque sea una opción a tener en cuenta en otro tipo de servicios, como podría ser la autenticación a distancias muy pequeñas de manera segura. Podrían complementarse.

---

## RFID

Las etiquetas *RFID* permiten, mediante señales de radio, la identificación del objeto que se encuentre asociado a la misma. Es una tecnología muy parecida a *NFC*, usándose principalmente para el control de suministros e inventarios en almacenes. La distancia de lectura varía entre 1 y 10 m.



Ilustración 3: Escaneo tarjeta RFID [12]



Existen dos tipos de *RFID tags*[13]. Por un lado se encuentra el *tag* pasivo, que no dispone de batería y se alimenta del lector de códigos, con un alcance igual al de *NFC*, y por el otro, el activo que sí que dispone de batería y su rango es mucho mayor (unos 100 metros).

La ventaja principal de *BLE*, a pesar de que es, en el fondo, muy parecido a una etiqueta *RFID* activa, reside en la tecnología *Bluetooth*. Gracias al uso de este estándar, los dispositivos *BLE* pueden ser localizados por cualquier *smartphone* que disponga de *Bluetooth 4.0*. Además, la inversión en investigación es mucho mayor y la implantación de dispositivos *Bluetooth* está mucho más generalizada.

---

## HARDWARE Y APIS DE FABRICANTES

Existen multitud de empresas que desarrollan dispositivos *iBeacon*, entre ellas las más famosas son, *Estimote*, *Gelo*, *Kontakt.io* y *Gimbal*.

La mayoría de estas empresas extienden la funcionalidad de sus dispositivos a través del desarrollo de un API propia, que usa como base la propia API de *Apple*. Por ejemplo, *Estimote* incorpora un sensor de temperatura y otro de movimiento. Aparte de software, también implementan servicios, como el registro de *beacons* en la nube, o cuestiones relacionadas con la seguridad.

En cuanto a precios, la mayoría suele rondar los 20 o 30 \$ por dispositivo y siempre enfocado a la investigación y desarrollo de aplicaciones. En el caso de implantar una solución que necesite gran cantidad de unidades, el precio desciende considerablemente.

Después de investigar las posibles opciones, se ha tomado la decisión de utilizar para el desarrollo los *beacons* de *Estimote*[14]. Los motivos son los siguientes:

- ✓ El motivo principal es la comunidad y documentación disponible. Según la propia página de *Estimote*, 30.000 desarrolladores se encuentran desarrollando con este tipo de *iBeacons*.
- ✓ Cuentan con un mecanismo de modificación de los *UUIDs* de los dispositivos para personalizarlos según las necesidades del negocio.
- ✓ Cuentan con un API que permite extender la funcionalidad de los dispositivos (temperatura, acelerómetro).
- ✓ Realizan periódicamente actualizaciones en el SDK, en la aplicación móvil y en el firmware de los dispositivos.





- ✓ Incorporan mejoras de seguridad y sincronización con la nube que permitirán un desarrollo más seguro y profesional. Además permiten la creación y visualización de estadísticas en su sitio web.

Aunque se haya elegido una marca concreta de *iBeacons*, el proyecto se realizará de la forma más genérica posible, usando el *framework* propio de *Apple* para la comunicación entre el dispositivo y el *beacon*.



## LIMITACIONES DE IBEACON

---

### SEGURIDAD

Uno de problemas principales que aparecen a la hora pensar en implantar un sistema monitorizado por *iBeacons* es la seguridad. Estos dispositivos emiten unas señales que pueden ser leídas por cualquier dispositivo compatible, lo que nos lleva al primer problema.

Si cualquiera puede saber cuáles son los valores *UUID*, *major* y *minor* de nuestro *iBeacon*, puede perfectamente crear su propia aplicación y suplantar al *beacon* en cuestión. Pero dentro de lo que cabe, en principio, este tipo de problemas no afectarían a nuestro sistema.

El problema principal que me he encontrado es la modificación de los valores del *beacon* (*uuid*, *major*, *minor*). El *framework* de *Apple* no permite la modificación de estos valores, pero el fabricante de los *beacons* que he usado como ejemplo, *Estimote*, ha ampliado el *framework* para poder cambiar distintos parámetros. Si un dispositivo *IOS* desea modificar algún dato del *beacon*, deberá usar la aplicación nativa del fabricante y modificarlos desde su interfaz. En caso de que el terminal sea *Android*, el *firmware* de los *beacons* de *Estimote* permite la modificación de sus valores sin el uso de su propia aplicación, por tanto, cualquiera podrá modificar sus valores.

Para solucionar estos problemas, *Estimote* modificó su propio *framework* y asoció cada *beacon* a una cuenta de usuario. El problema se solucionó en *IOS*, ya que con su aplicación nativa se controla quién accede y quién puede modificar, pero en el caso de *Android*, a día de hoy, el problema sigue estando presente.

Al ser una tecnología muy novedosa, es normal que existan este tipo de inconvenientes hasta que madure un poco. Según la página web de *Estimote*, están trabajando para solucionar este tipo de incidencias.

En el caso de otros fabricantes supongo que, si desean poder modificar los datos que envían los *beacons*, deberán enfrentarse a problemas semejantes.

---

### PRECISIÓN

Aunque para el proyecto a desarrollar en principio no importa mucho los temas relativos a la precisión, sí que es interesante saber sus limitaciones.



Básicamente los *beacon* emiten señales basadas en *BLE* que tienen características muy parecidas a las de radio. Su característica principal es que pueden atravesar objetos, aunque esto provoca que su señal se atenúe bastante. Esto junto a las fluctuaciones en la intensidad con la que un *smartphone* recibe las señales de los *beacons*, provoca que sea muy difícil ajustar correctamente la distancia a la que se encuentra.

En el *framework* de *Apple*, no se proporciona un valor “distancia” entre el *beacon* y el dispositivo. En su lugar, se proporcionan cuatro rangos en los que se puede encontrar el *beacon*: “inmediatamente”, “cerca”, “lejos” o “desconocido”.

El rango “inmediatamente” significa que el *beacon* se encuentra a menos de un metro del móvil, “cerca” entre uno y tres metros, “lejos” mayor a tres metros y en caso de no encontrar un valor, mostrará “desconocido”.

Estos valores son proporcionados analizando la intensidad con la un móvil recibe la señal. Por supuesto, son totalmente orientativos, por ejemplo, podemos estar a menos de un metro y que muestre el valor “lejos” porque tenemos algún objeto que atenúa la señal.

Se realizaron pruebas de triangulación para estudiar si se podían usar a modo de *gps* en interiores, cuya aplicación mostraría la ubicación del móvil en todo momento. Debido a las fluctuaciones que puede sufrir la intensidad de la onda, se podían tomar valores muy dispares incluso estando fijos en un sitio. Por ejemplo, ubicando el *beacon* a unos diez metros del *smartphone*, se obtuvieron valores tan dispares con un rango de entre cinco y cuarenta metros, en un periodo breve de tiempo y con la señal estabilizada.

Por tanto, recibiendo valores tan dispares, fue imposible calcular una cifra que diese un valor representativo válido.

A modo de conclusión sobre la precisión de los *beacons*, decir que, la mayor precisión que podemos conseguir, cuyo valores son los más fiables, son los proporcionados por el *framework* de *Apple* y, posiblemente, por este motivo no proporcionan en su *framework* el valor “distancia”.

---

## DETECCIÓN DE DISPOSITIVOS

La detección de los *iBeacons* por parte de los *smartphones* se puede realizar de varias maneras y depende de numerosos factores.

La forma más eficaz de detectar los *beacons* es usando el método de escaneo que proporciona el *framework* de *Apple*. Este método, una vez iniciado, se dedica a buscar señales de *beacons* de manera ininterrumpida, hasta que el usuario cierra la aplicación o se ejecuta el método de parar. La principal ventaja consiste en que funciona muy bien para detectar constantemente los dispositivos cercanos,

pero su problema principal, si deseamos un funcionamiento óptimo, reside en que necesita que la aplicación se encuentre activa.

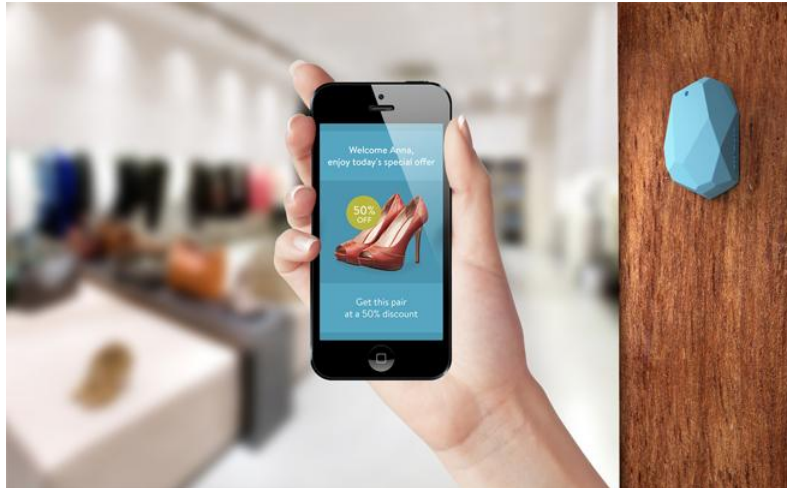


Ilustración 4: Detección iBeacon [15]

Otra forma de detectar *beacons* consiste en usar la entrada/salida de región, cuyo sistema funciona en segundo plano incluso con la aplicación parada, pero con limitaciones. Una limitación importante consiste en que para detectar que el usuario ha salido de una región, el Smartphone establece un tiempo por el cual no debe recibir ninguna señal e interprete que ya no se encuentra dentro. Por ejemplo, si un dispositivo no detecta ningún *beacon* en 10 segundos, pero a continuación recibe otra nueva, puede interpretar que en realidad no ha salido de la región. Esto sumado a las características del modo suspendido del móvil, que provocan que no siempre esté comprobando si tiene *beacons* cercanos, puede causar imprecisión en los tiempos de salida (pero si orientativos). En cambio, los tiempos de entrada sí proporcionan datos bastante precisos, porque en el momento de detectar un *beacon*, siempre que no estemos dentro de una región, interpretará al instante la entrada en la misma.

Por último, si el *smartphone* no tiene activa una conexión a internet y se encuentra suspendido, la frecuencia de las comprobaciones serán mucho menores que en el caso de tener alguno de estos sistemas activos.



## CASOS DE UTILIZACIÓN

En este apartado se van a mostrar algunas de las posibles aplicaciones que se podrían desarrollar con esta tecnología. Muchos de estos ejemplos se pueden complementar con la prueba de concepto realizada, de manera transparente al usuario.

- Información en lugares de interés cultural (zoo, museo, parque de atracciones).
- Transporte público: localización sin cobertura en lugares como el metro.
- Publicidad personalizada. Por ejemplo, un escaparate de una tienda que muestre un video corto con modelos con ropa afín.
- Recomendaciones de productos en función del comportamiento. Por ejemplo, en un centro comercial, si te mueves mucho por una zona reconocer sus gustos y mostrarle información acorde.
- Mensajes de bienvenida (por ejemplo con ofertas) en una discoteca, hotel, etc...
- Ayuda a ciegos para moverse por edificios.
- Localización de productos en tiendas o grandes superficies.
- Ofertas en restaurantes, por ejemplo si eres celíaco o vegetariano.
- Al entrar en una habitación de hotel, recibir un mensaje *push* con los servicios, actividades, horarios, precio de los productos del mini bar.
- Navegación por interiores. Por ejemplo, en el caso de un parking, te guía hasta tu posición, pero no se muestra todo el mapa públicamente. Existen servicios como *Google Maps* que implementan rutas por los interiores de algunos edificios. Esto puede provocar problemas de seguridad en algunos casos, debido a que la accesibilidad a estos planos es pública. En el caso de usar *beacons*, el mapeo permanecería oculto.

## EJEMPLOS DE IMPLANTACIÓN

A continuación se expondrán brevemente algunos casos reales en los que ya ha sido implantada la tecnología *iBeacon* y algunos resultados obtenidos.

### MCDONALD'S

*McDonald's* ha desarrollado un estudio sobre tecnología *iBeacon* en sus locales de la ciudad de Columbia en Estados Unidos.

Dicho estudio consiste en enviar ofertas personalizadas a los consumidores mediante *beacons*, a sus dispositivos. Estas ofertas se basan en su localización, por ejemplo, si se desea promocionar un producto en concreto en un determinado establecimiento.

Las conclusiones del estudio indican que se produjo un incremento en las ventas de los productos promocionados en los locales que contaban con esta tecnología. En concreto, alrededor de un 8%[16].

### JIVR

*Jivr* [17] es una bicicleta eléctrica pensada para su uso en la ciudad y de tipo plegable. Su diferencia principal es la implantación de tecnología *iBeacon* para comunicar la bicicleta a todos los dispositivos compatibles que se encuentre a su paso, por ejemplo, balizas u otros vehículos. También cuenta con un espacio en el manillar para colocar el *smartphone* y poder así comunicarse con los *beacons* situados en la bicicleta.



Ilustración 5: Diseño Jivr [17]



---

## AENA

*Aena*[18] se une también al desarrollo de prototipos y ha implementado un servicio basado en *iBeacons* en los aeropuertos de Madrid y Barcelona.

El servicio consistirá en una aplicación móvil que proporcionará información personalizada a los usuarios acerca de su vuelo, documentación para embarcar, ubicación del vehículo en el parking, etc... Todo ello en tiempo real.

---

## MLB

La liga de beisbol *MLB (Major League Baseball)* [19] de Estados Unidos, ha implantado en un total de veinte estadios la tecnología *iBeacon*, con un total de cien dispositivos.

La idea de los desarrolladores de esta aplicación es dar más oportunidades de interactuar y aprender sobre los estadios y equipos donde se juega.

---

## FITUR

Durante la edición de la Feria Internacional de Turismo (*Fitur*) de 2015, se presentó una aplicación sobre el Quijote[20]. Esta aplicación permite la localización en interiores del turista, avisando de eventos mediante notificaciones o en la pantalla del móvil.

Algunas de las funcionalidades de esta aplicación son, por ejemplo, mostrar información relativa a un cuadro cuando el usuario se acerca a él o guiar al turista hasta un punto de información dónde puede ser atendido.

---

## SMARTH COSTA DEL SOL

El ayuntamiento de Rincón de la Victoria[21] ha propuesto una iniciativa basada también en esta tecnología. En concreto los servicios que desean implantar están orientados a fomentar e impulsar los servicios turísticos de la zona.

Por ejemplo, se implantarán sensores en las playas, para saber el estado del agua o la ocupación, puntos de interés para el viajero, información relacionada con los comercios turísticos de la zona, etc...

Actualmente el proyecto se encuentra en una fase de análisis y esperan implantarlo para 2016.







## DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA

La monitorización de la superficie se va a realizar colocando pequeñas balizas denominadas *beacons*, que funcionan con tecnología *Bluetooth* de baja energía (*BLE*, también conocido como *BT Smart*), en las entradas y salidas del comercio, así como, lugares estratégicos.

La configuración, tanto del rango del *beacon* como del intervalo entre emisiones de la señal, se realizará a través de una *aplicación* creada por el desarrollador de los dispositivos, ya que se trata de la única forma disponible para modificar esos valores. La sincronización con la aplicación de monitorización se realizará captando los valores que identifican a los *beacons* y definiendo qué regiones debe escuchar la aplicación.

En la parte cliente se desarrollará una aplicación para dispositivos con sistema operativo *IOS*. Esta aplicación será capaz de captar la señal enviada por los *beacons* y enviar información servidor, en formato *JSON*. Existirá la posibilidad de trabajar *offline*, volcando la información al servidor una vez esté conectado a la red. Esta información reflejará el *id* del *beacon* que ha detectado, la fecha y hora y el *UUID* del dispositivo. También detectará si se trata de un evento producido por la entrada o salida en una región. Nos situaremos dentro de una región cuando detectemos un *beacon* y saldremos cuando dejemos de recibir la señal.

En la parte servidor, por un lado tiene que ser capaz de recibir, procesar y guardar la información recibida por el dispositivo del cliente. Por otro lado, se creará una interfaz de usuario para mostrar los resultados obtenidos. Todo ello se realizará usando las herramientas *PHP* y *MySQL* para la parte servidora y *HTML*, *CSS* y *Bootstrap* para la interfaz de usuario.

Existen multitud de alternativas a *PHP* y *MySQL*. En el caso de *PHP* se encuentra por ejemplo *Python*, pero se ha escogido *PHP* por su extensa documentación. También cumple un propósito parecido *Java*, pero ha sido descartado debido a que su uso está más orientado a ambientes empresariales.

En el caso de *MySQL*, existen alternativas de código abierto como *MariaDB*, *PostgreSQL* y *SQLite*. Al final se ha optado por *MySQL* por su velocidad, bajo consumo y extensa documentación.

En la parte cliente se ha elegido *IOS* en lugar de *Android* por el tipo de tecnología a implementar. El desarrollo con *iBeacons* está mucho más extendido y por tanto existe más documentación y soluciones, con el sistema operativo *IOS* (lo cual es razonable debido a que tanto *iOS* como *iBeacon* han sido diseñado por la misma compañía).

A continuación se muestra un gráfico dónde se muestra como quedaría la estructura de la propuesta.

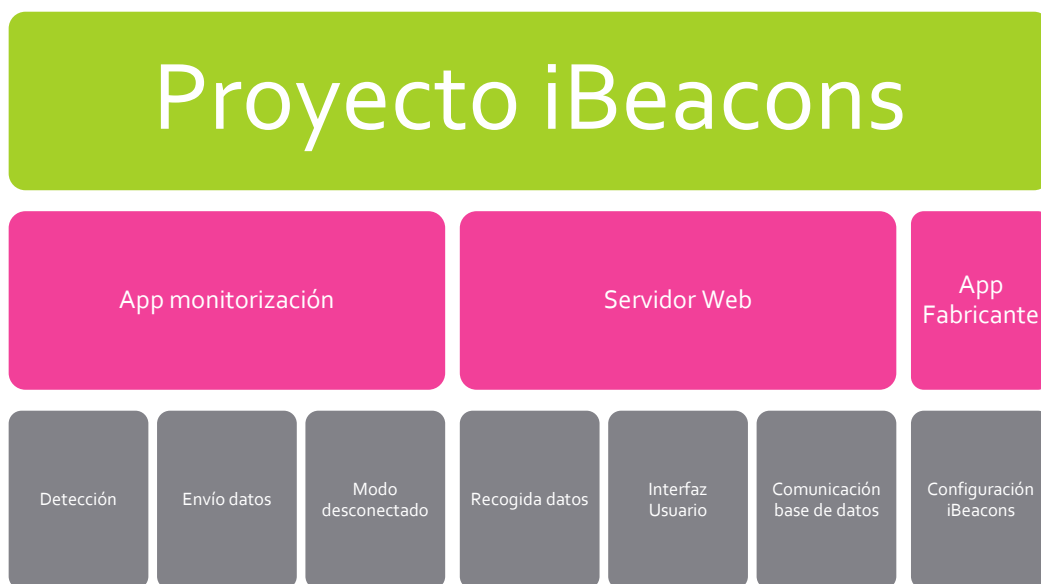


Ilustración 6: Gráfico arquitectura sistema

Por otro lado el uso de *JSON* ayuda significativamente a la hora de procesar la información en la parte cliente, sobre todo a la hora de crear estructuras como diccionarios sin necesidad de procesar los datos.

En el caso de *XML* al contener etiquetas, antes de pasar la información a una estructura de datos, habría que filtrar eliminando todos los *tags* (etiquetas) y organizando la información.

Por último, se hablará sobre la distribución de los números que identifican cada *beacon*. Cada baliza consta de *uuid*, *major* y *minor*. El *uuid* no es modificable y deberá tener el mismo valor para todos los dispositivos de la aplicación. Tanto el *major* como el *minor* constan de cinco dígitos, cuyo valor máximo es el de 65535. Dada esta limitación se expone la siguiente representación:

- *Major*: Los dos primeros dígitos representan el *id* del *Beacon*. Los tres últimos el *id* del *Estudio*. Por ejemplo, *beacon* número 3 del estudio 7: *major* = 03007.
- *Minor*: Los tres primeros dígitos representan el *id* de la tienda. Los dos últimos el *id* de la posición. Ejemplo; *beacon* de la zona de panadería con *id* 15 y la tienda con *id* 32: *minor* = 03215.

## ANÁLISIS DE LA PROPUESTA

### DEFINICIÓN DE CASOS DE USO

A la hora de diseñar una aplicación de este calibre, es necesario identificar una serie de casos de uso, para facilitar las labores de especificación de requisitos, así como diseñar los diagramas que representan la arquitectura tanto lógica, como física del proyecto, como se aprecia en el apartado *“Desarrollo e Implementación de la propuesta”*.

Primero se muestra el diagrama de casos de uso para después especificar de manera más concreta cada caso. Señalar que se han agrupado los casos de uso “Dar de alta”, “Dar de baja” y “Modificar” así como los casos de uso correspondientes con las visualizaciones de las secciones, porque todos van a pertenecer al rol administrador, salvo el caso de “Visualizar estadísticas”, como se muestra en el diagrama.

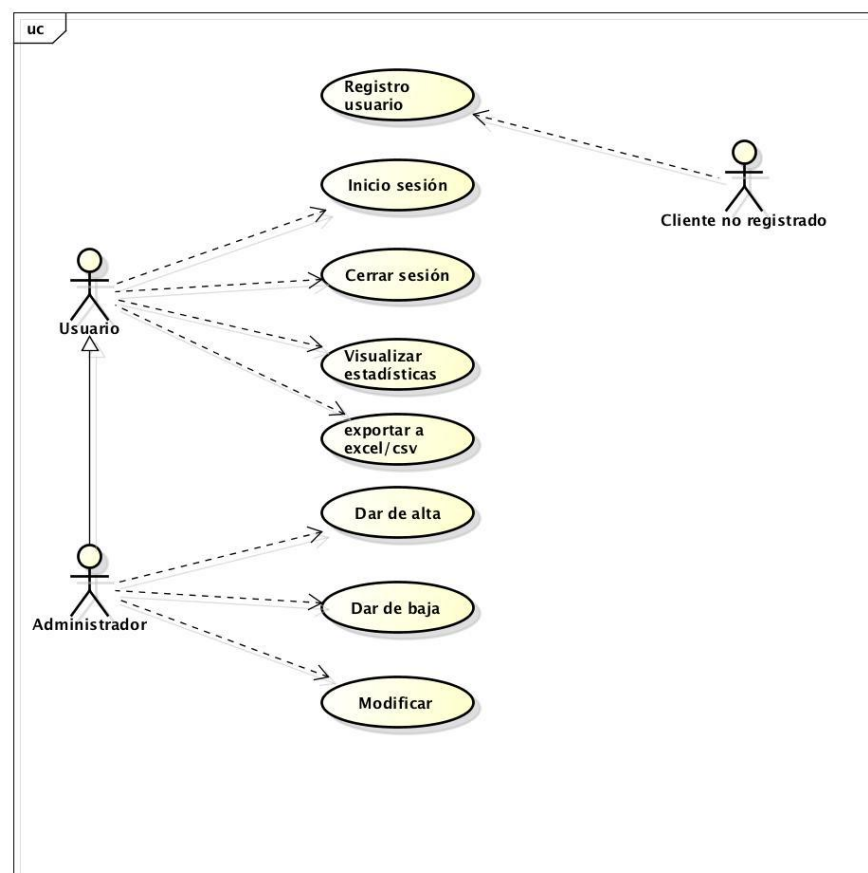


Ilustración 7: Diagrama casos de uso



A continuación, se describen los casos de uso del sistema.

La tabla que se muestra a continuación sirve como plantilla para los casos de uso identificados:

Identificador	
Descripción	
Pre-condiciones	
Escenario	
Post-condiciones	

Tabla 3: Plantilla de casos de uso

A continuación se incluye una breve descripción de los campos que componen la tabla:

- **Identificador:** Nombre identificativo del caso de uso. Se utilizará la nomenclatura CU-<XX>, donde CU hace referencia a 'caso de uso' y <XX> será un código numérico desde 01 a 99.
- **Descripción:** Indica en qué consiste dicho caso de uso.
- **Pre-condiciones:** Condiciones iniciales que se deben cumplir para realizar el caso de uso.
- **Escenario:** Conjunto de pasos a seguir para realizar el caso de uso.
- **Post-condiciones:** Estado del sistema tras realizarse el caso de uso.

Los casos de uso identificados para la propuesta son los siguientes:

CU-01	
Descripción	Inicio de sesión.
Pre-condiciones	El usuario se encuentra en la pantalla de inicio de sesión.
Escenario	<ol style="list-style-type: none"><li>1. El usuario introduce sus credenciales y pide la entrada al sistema.</li><li>2. El sistema comprueba si la información es correcta, analiza el tipo de rol y muestra el menú principal.</li></ol>
Post-condiciones	El usuario se encuentra autenticado y puede visualizar el menú principal.

Tabla 4: CU-01 - Inicio de sesión

CU-02	
Descripción	Registro de usuario.
Pre-condiciones	El usuario se encuentra en la pantalla de registro de usuario.
Escenario	<ol style="list-style-type: none"><li>1. El usuario rellena el formulario con la información correspondiente.</li><li>2. El sistema registra un nuevo usuario.</li></ol>



Post-condiciones	Se ha creado un nuevo usuario.
------------------	--------------------------------

Tabla 5: CU-02 – Registro de usuario

CU-03	
Descripción	Cerrar sesión.
Pre-condiciones	El usuario se encuentra autenticado y con una sesión activa dentro del sistema.
Escenario	<ol style="list-style-type: none"><li>1. El usuario inicia el cierre de sesión.</li><li>2. El sistema cierra la sesión del usuario y redirige al menú de login.</li></ol>
Post-condiciones	El usuario se encuentra fuera de la sesión y en el menú de login.

Tabla 6: CU-03–Cerrar sesión

CU-04	
Descripción	Visualización de estadísticas.
Pre-condiciones	El usuario ha iniciado sesión.
Escenario	<ol style="list-style-type: none"><li>1. El usuario selecciona el estudio del cual se mostrarán los resultados.</li><li>2. El sistema muestra un listado con todos los datos asociados al estudio.</li></ol>
Post-condiciones	El usuario puede visualizar las estadísticas asociadas al estudio.

Tabla 6: CU-04–Visualización estadísticas

CU-05	
Descripción	Visualizar el listado de iBeacons.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú principal.
Escenario	<ol style="list-style-type: none"><li>1. El usuario selecciona la pestaña de administración de iBeacons.</li><li>2. El sistema muestra un listado con todos los iBeacons asociados .</li></ol>
Post-condiciones	El usuario puede visualizar el listado con todos los iBeacons.

Tabla 7: CU-05 – Visualizar iBeacons

CU-06	
Descripción	Dar de alta un iBeacon.



Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de iBeacons.
Escenario	<ol style="list-style-type: none"><li>1. El usuario inicia la creación de un nuevo iBeacon.</li><li>2. El usuario rellena la información correspondiente.</li><li>3. El sistema registra un nuevo iBeacon.</li></ol>
Post-condiciones	Se ha creado un nuevo iBeacon.

Tabla 8: CU-06–Dar de alta iBeacon

CU-07	
Descripción	Dar de baja un iBeacon.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de iBeacons.
Escenario	<ol style="list-style-type: none"><li>1. El usuario inicia la baja de un iBeacon.</li><li>2. El usuario confirma que desea eliminar dicho iBeacon.</li><li>3. El sistema elimina el iBeacon.</li></ol>
Post-condiciones	Se ha eliminado un iBeacon.

Tabla 9: CU-07 – Dar de baja iBeacon

CU-08	
Descripción	Modificar un iBeacon.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de iBeacons.
Escenario	<ol style="list-style-type: none"><li>1. El usuario inicia la modificación de un iBeacon.</li><li>2. El usuario rellena el formulario con la información correspondiente.</li><li>3. El sistema edita el iBeacon.</li></ol>
Post-condiciones	La información relativa a un iBeacon muestra los nuevos valores editados.

Tabla 10: CU-08–Modificar iBeacon

CU-09	
Descripción	Visualizar el listado de estudios.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú principal.
Escenario	<ol style="list-style-type: none"><li>3. El usuario selecciona la pestaña de administración de estudios.</li><li>4. El sistema muestra un listado con todos los estudios asociados.</li></ol>
Post-condiciones	El usuario puede visualizar el listado con todos los estudios.

Tabla 11: CU-09–Visualizar estudios



CU-10	
Descripción	Dar de alta un estudio.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de estudios.
Escenario	4. El usuario inicia la creación de un nuevo estudio. 5. El usuario rellena la información correspondiente. 6. El sistema registra un nuevo estudio.
Post-condiciones	Se ha creado un nuevo estudio.

Tabla 12: CU-10 – Dar de alta estudio

CU-11	
Descripción	Dar de baja un estudio.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de estudios.
Escenario	4. El usuario inicia la baja de un estudio. 5. El usuario confirma que desea eliminar dicho estudio. 6. El sistema elimina el estudio.
Post-condiciones	Se ha eliminado un estudio.

Tabla 13: CU-11 –Dar de baja estudio

CU-12	
Descripción	Modificar un estudio.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de estudios.
Escenario	4. El usuario inicia la modificación de un estudio. 5. El usuario rellena el formulario con la información correspondiente. 6. El sistema edita el estudio.
Post-condiciones	La información relativa a un estudio muestra los nuevos valores editados.

Tabla 14: CU-12–Modificar estudio

CU-13	
Descripción	Visualizar el listado de usuarios.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú principal.
Escenario	5. El usuario selecciona la pestaña de administración de usuarios. 6. El sistema muestra un listado con todos los usuarios asociados .



Post-condiciones	El usuario puede visualizar el listado con todos los usuarios.
------------------	--

Tabla 15: CU-13–Visualizar usuarios

<b>CU-14</b>	
Descripción	Dar de baja un usuario.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de usuarios.
Escenario	<ol style="list-style-type: none"><li>7. El usuario inicia la baja de un usuario.</li><li>8. El usuario confirma que desea eliminar dicho usuario.</li><li>9. El sistema elimina el usuario.</li></ol>
Post-condiciones	Se ha eliminado un usuario.

Tabla 16: CU-14–Dar de baja usuario

<b>CU-15</b>	
Descripción	Modificar un usuario.
Pre-condiciones	El usuario ha iniciado sesión como administrador y situado en el menú de gestión de usuarios.
Escenario	<ol style="list-style-type: none"><li>7. El usuario inicia la modificación de un usuario.</li><li>8. El usuario rellena el formulario con la información correspondiente.</li><li>9. El sistema edita el usuario.</li></ol>
Post-condiciones	La información relativa a un usuario muestra los nuevos valores editados.

Tabla 17: CU-15–Modificar usuario

<b>CU-16</b>	
Descripción	Exportar datos a excel.
Pre-condiciones	El usuario ha iniciado sesión.
Escenario	<ol style="list-style-type: none"><li>1. El usuario selecciona el estudio del cual se mostrarán los resultados.</li><li>2. El sistema muestra un listado con todos los datos asociados al estudio.</li><li>3. El usuario selecciona exportar los datos a Excel y se descargará el archivo en formato .xls</li></ol>
Post-condiciones	El usuario se ha descargado el documento Excel con los resultados del estudio.

Tabla 18: CU-16–Exportar a excel





CU-17	
Descripción	Exportar datos a csv.
Pre-condiciones	El usuario ha iniciado sesión.
Escenario	<ol style="list-style-type: none"><li>1. El usuario selecciona el estudio del cual se mostrarán los resultados.</li><li>2. El sistema muestra un listado con todos los datos asociados al estudio.</li><li>3. El usuario selecciona exportar los datos a csv y se descargará el archivo en formato .csv</li></ol>
Post-condiciones	El usuario se ha descargado el documento csv con los resultados del estudio.

Tabla 19: CU-17–Exportar a csv



## REQUISITOS DEL SISTEMA

En este apartado se incluyen todos los requisitos, funcionales y no funcionales, para el software que se plantea en este trabajo. Dichos requisitos son la base sobre la que se realiza el desarrollo de la aplicación.

Cada uno de los requisitos vendrá definido por una tabla como la siguiente:

Identificador					
Descripción					
Verificabilidad	<input type="checkbox"/> Alta	Necesidad	<input type="checkbox"/> Esencial	Prioridad	<input type="checkbox"/> Alta
	<input type="checkbox"/> Media		<input type="checkbox"/> Deseable		<input type="checkbox"/> Media
	<input type="checkbox"/> Baja		<input type="checkbox"/> Opcional		<input type="checkbox"/> Baja

Tabla 20: Plantilla de requisitos

A continuación se incluye una breve descripción de los campos que componen la tabla:

- **Identificador:** Este campo corresponde al identificador del requisito. Dicho identificador será único e inconfundible, de tal forma que cada requisito sea identificable sin posibilidad de error. Cada uno de los identificadores seguirá la siguiente nomenclatura:
  - **Requisitos funcionales:** RF-<XXX>-<YYY>, siendo <XXX>el valor app (referente a la aplicación móvil) o serv (referente al servicio web) y <YYY>un número entre 001 y 999.
  - **Requisitos no funcionales:** RNF-<XXX>-<YYY>, siendo <XXX>el valor app (referente a la aplicación móvil) o serv (referente al servicio web) y <YYY>un número entre 001 y 999.
- **Descripción:** Campo que incluye una descripción del requisito en cuestión.
- **Necesidad:** Este campo indica la necesidad de incorporar el requisito en el sistema. Los posibles valores son:
  - **Esencial:** El requisito debe introducirse obligatoriamente dentro del sistema desarrollado.
  - **Deseable:** La incorporación de estos requisitos en el sistema puede darse en función del desarrollo del mismo.
  - **Opcional:** La incorporación del requisito en el sistema es opcional.



- **Verificabilidad:** Indica la posibilidad de comprobar que el requisito se haya incorporado al sistema. Los posibles valores son:
  - **Alta:** Se puede comprobar de forma sencilla y sin ningún tipo de dudas que el requisito está en el sistema final.
  - **Media:** La comprobación de que el requisito ha sido incorporado en el sistema no tiene una dificultad muy elevada.
  - **Baja:** Es difícil o imposible comprobar que el requisito ha sido introducido en el sistema final.
- **Prioridad:** Este campo indica el grado de prioridad con el que debe ser resuelto un requisito. Los posibles valores son:
  - **Alta:** El diseño e implementación del requisito es de carácter prioritario.
  - **Media:** El requisito se debe diseñar e implementar con prioridad media.
  - **Baja:** El requisito se debe diseñar e implementar con prioridad baja.

## REQUISITOS FUNCIONALES

A continuación se listan todos los requisitos funcionales para la aplicación.

RF-app-001					
Descripción		La aplicación detectará la entrada o salida del rango de los beacons dados de alta en el servidor.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 21: RF-app-001

RF-app-002					
Descripción		Cada vez que el usuario entre o salga del rango de un beacon, la aplicación enviará al servidor información relativa al UID del dispositivo, al beacon asociado, fecha, hora y si entró o salió de la región (siempre que exista conexión de red).			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 22: RF-app-002



RF-app-003					
Descripción		Una vez instalada la aplicación en el terminal del cliente, la aplicación mostrará un mensaje en el que se informe que se ha instalado satisfactoriamente y las instrucciones para que funcione correctamente.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja

Tabla 23: RF-app-003

RF-app-004					
Descripción		La detección de entrada o salida de las regiones deberá realizarse en segundo plano, de la manera más transparentemente posible al usuario.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 24: RF-app-004

RF-app-005					
Descripción		La aplicación detectará si el dispositivo tiene conexión a internet o no. En caso negativo, los datos se enviarán al servidor en diferido.			
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 25: RF-app-005

RF-serv-001	
Descripción	Tanto elroladministrador como el usuario tendrán acceso a la sección de estadísticas.



Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
-----------------	---	-----------	--	-----------	---

Tabla 26: RF-serv-001

RF-serv-002					
Descripción		El administrador tendrá acceso a la sección de gestión de Beacons.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 27: RF-serv-002

RF-serv-003					
Descripción		El administrador tendrá acceso a la sección de gestión de estudios.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 28: RF-serv-003

RF-serv-004					
Descripción		El administrador tendrá acceso a la sección de gestión de usuarios.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 29: RF-serv-004

RF-serv-005	
Descripción	La sección de estadísticas mostrará los valores relativos a la monitorización enviados por los dispositivos móviles.



<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
------------------------	---	------------------	--	------------------	---

Tabla 30: RF-serv-005

RF-serv-006					
<b>Descripción</b>		La sección de gestión de iBeacons mostrará una lista con todos los iBeacons existentes, pudiendo generar, editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).			
<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 31: RF-serv-006

RF-serv-007					
<b>Descripción</b>		La sección de gestión de estudios mostrará una lista con todos los estudios existentes, pudiendo generar, editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).			
<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 32: RF-serv-007

RF-serv-008	
<b>Descripción</b>	La sección de gestión de usuarios mostrará una lista con todos los usuarios existentes, pudiendo editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).



Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja
-----------------	---	-----------	--	-----------	---

Tabla 33: RF-serv-008

RF-serv-009					
Descripción		La opción de añadir iBeacon mostrará un formulario en la página que permitirá al usuario introducir valores para cada campo del iBeacon.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 34: RF-serv-009

RF-serv-010					
Descripción		La opción de editar iBeacon mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho iBeacon.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 35: RF-serv-010

RF-serv-011					
Descripción		La opción de eliminar iBeacon permitirá, previa confirmación del usuario, suprimir el iBeacon de la base de datos.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 36: RF-serv-011



RF-serv-012					
Descripción		La opción de añadir estudio mostrará un formulario en la página que permitirá al usuario introducir valores para cada campo del estudio.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 37: RF-serv-012

RF-serv-013					
Descripción		La opción de editar estudio mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho estudio.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 38: RF-serv-013

RF-serv-014					
Descripción		La opción de eliminar estudio permitirá, previa confirmación del usuario, suprimir el estudio de la base de datos.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 39: RF-serv-014

RF-serv-015	
Descripción	La opción de editar usuario mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho usuario.





<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
------------------------	---	------------------	--	------------------	---

Tabla 40: RF-serv-015

RF-serv-016					
<b>Descripción</b>		La opción de eliminar usuario permitirá, previa confirmación del usuario, suprimir el usuario de la base de datos.			
<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 41: RF-serv-016

RF-serv-017					
<b>Descripción</b>		La interfaz de usuario deberá adaptarse a dispositivos móviles.			
<b>Verificabilidad</b>	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	<b>Prioridad</b>	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja

Tabla 42: RF-serv-017

RF-serv-018					
<b>Descripción</b>		Las contraseñas de los usuarios del servicio serán encriptadas con el algoritmo SHA256.			
<b>Verificabilidad</b>	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	<b>Necesidad</b>	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	<b>Prioridad</b>	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 43: RF-serv-018

RF-serv-019					
<b>Descripción</b>		El sistema podrá descargar los resultados obtenidos de los			



estudios en los formatos Excel y csv.					
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 44: RF-serv-019



## REQUISITOS NO FUNCIONALES

A continuación se listan todos los requisitos no funcionales para la aplicación.

RNF-app-001					
Descripción		La aplicación podrá ejecutarse en el sistema operativo IOS versión 7.1.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 45: RNF-app-001

RNF-app-002					
Descripción		La aplicación funcionará con Bluetooth 4.0.			
Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 46: RNF-app-002

RNF-app-003					
Descripción		La aplicación será capaz de mantener un rendimiento óptimo en dispositivos de diferentes características.			
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja

Tabla 47: RNF-app-003

RNF-app-004					
Descripción		La aplicación requerirá un mínimo de 2 megas de espacio en disco			



		duro para su instalación.			
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja

Tabla 48: RNF-app-004

RNF-serv-001					
Descripción		El servicio web se podrá ejecutar desde cualquiera de los siguientes navegadores: Chrome, Firefox, Internet Explorer y Safari.			
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja

Tabla 49: RNF-serv-001

RNF-serv-002					
Descripción		Los datos del servicio web sólo podrán ser editados por los usuarios autenticados que tengan permiso para ello.			
Verificabilidad	<input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja

Tabla 50: RNF-serv-002



## DISEÑO E IMPLEMENTACIÓN DE LA PROPUESTA

En este apartado se ofrece el diseño físico-lógico del proyecto, así como una serie de pequeñas secciones donde se profundizará en las diferentes tareas de implementación realizadas a lo largo del desarrollo del trabajo.

### DISEÑO LÓGICO

En este apartado se procede a profundizar en el diseño lógico propuesto. Para ello, y partiendo de la base que conforman los casos de uso y los requisitos definidos, se aportará la siguiente información:

- **Diagrama de paquetes:** Establece la distribución lógica de funcionalidades, así como sus dependencias, tanto de la parte servidora como de la aplicación móvil.
- **Diagrama de despliegue:** Detalla la disposición física del sistema completo a implementar, detallando la comunicación entre los diferentes nodos y los componentes ejecutados.
- **Diagrama de modelo de datos:** Detalla la disposición lógica de los datos en su representación en la base de datos.

## DIAGRAMA DE PAQUETES

A continuación se muestra el diagrama de paquetes para el proyecto a desarrollar.

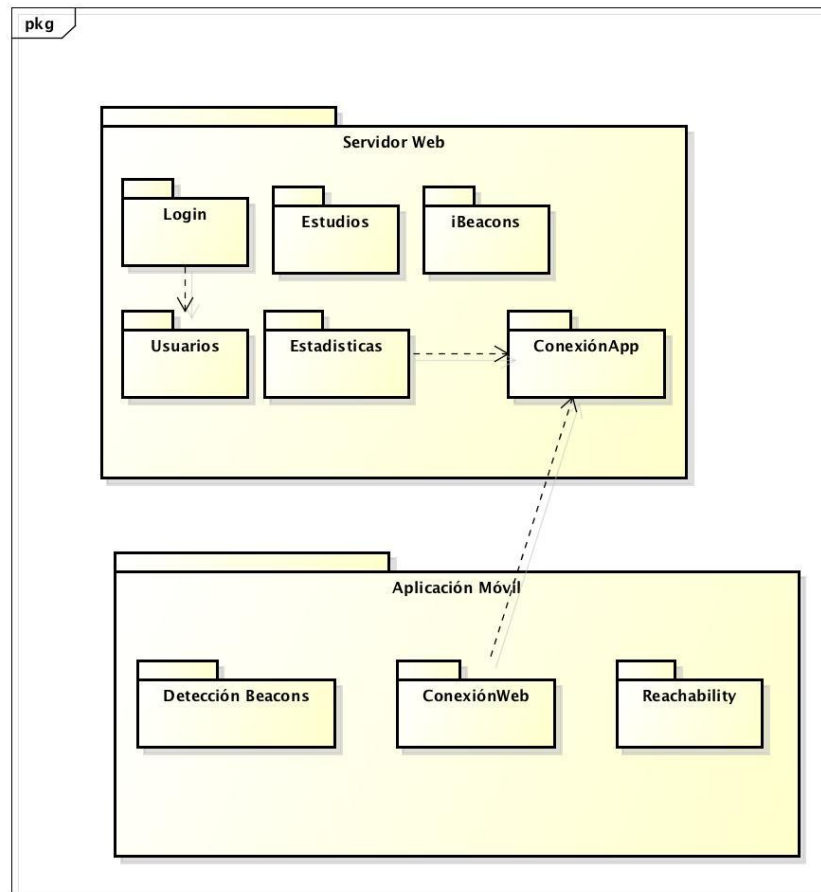


Ilustración 8: Diagrama de paquetes



## DIAGRAMA DE DESPLIEGUE

El siguiente diagrama muestra la estructura física del sistema a implementar así como sus conexiones.

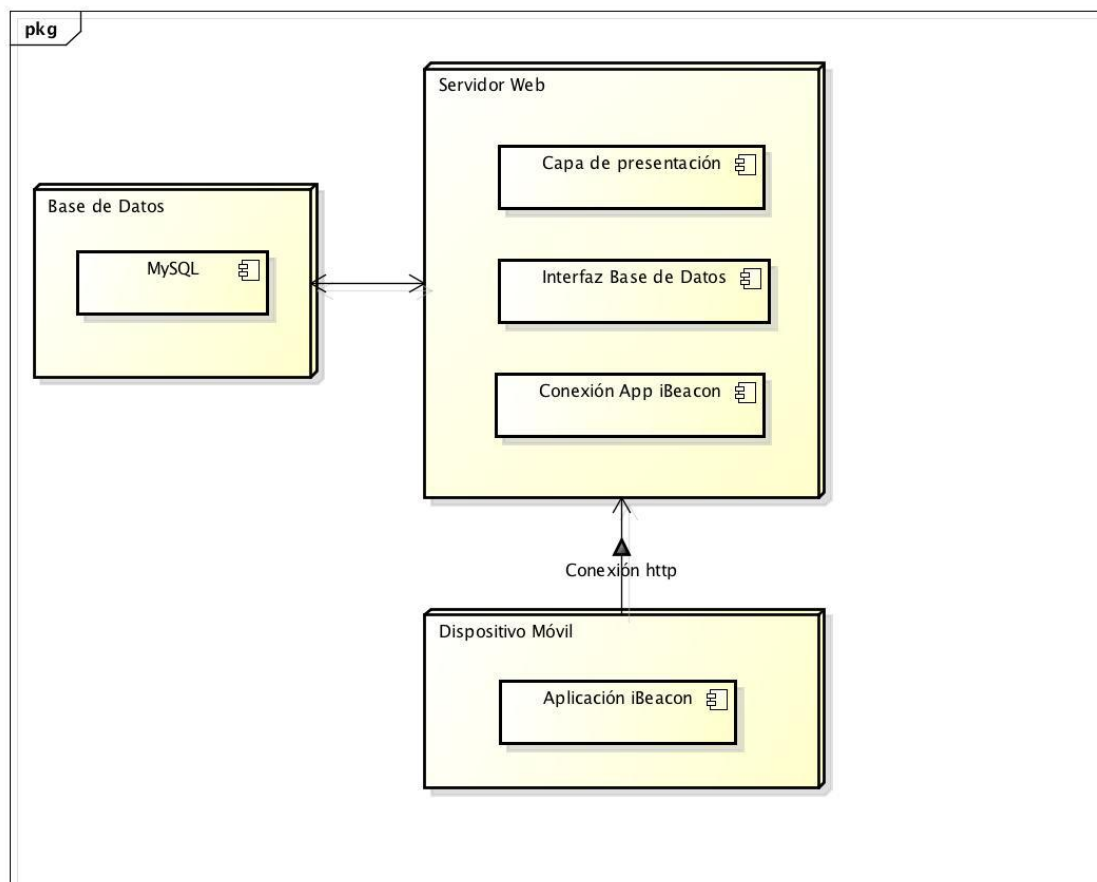


Ilustración 9: Diagrama de despliegue



## DIAGRAMA DE MODELO DE LOS DATOS

En este diagrama permite mostrar la estructura lógica del almacenamiento de los datos así como sus relaciones.

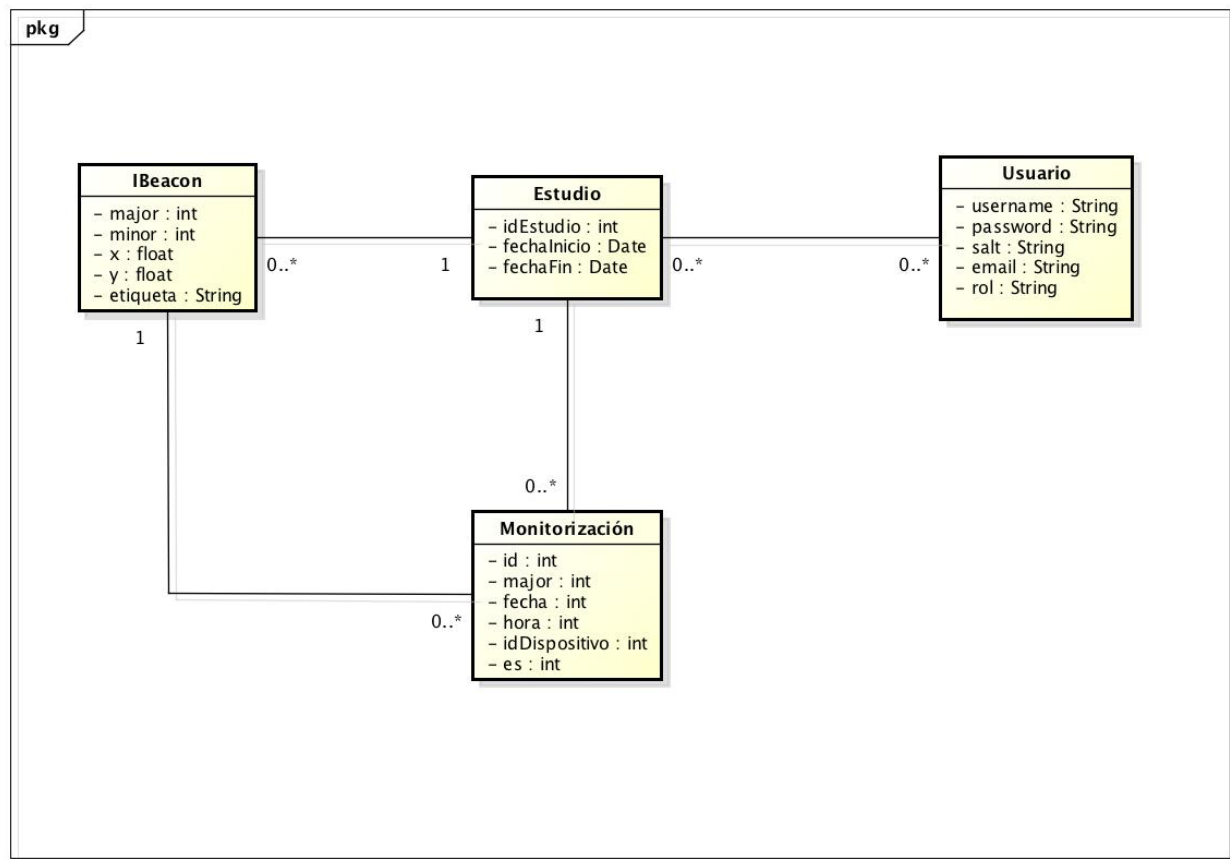


Ilustración 10: Diagrama modelo de datos





## DISEÑO FÍSICO

En esta sección se expone el diseño físico de la propuesta, en el que se especifican las características de los componentes del sistema requeridos para poner en práctica lo mostrado en el diseño lógico.

Se hará una división, mostrando primero los elementos requeridos para las labores de desarrollo y pruebas y después se especificará qué necesita un usuario para la ejecución de la misma.

Los elementos requeridos para las labores de desarrollo y pruebas serán los siguientes:

- Ordenador con sistema operativo *OS X*. Es necesario este sistema operativo para utilizar el entorno de desarrollo *Xcode*, el cual permite el desarrollo para dispositivos *IOS*.
- Un dispositivo con sistema operativo *IOS* y *Bluetooth 4.0*. En el caso del desarrollo, en principio, *Xcode* permite ejecutar la aplicación en simuladores, pero no dispone de servicios como *Bluetooth*, así que, para poder realizar pruebas necesitamos un terminal con las características descritas.
- Un servidor que sea visible desde la red. Este requisito es imprescindible para la comunicación entre el *smartphone* y la aplicación web, ya que, una vez que el dispositivo detecta el *beacon*, necesita de una dirección de un servidor para poder guardar la información en la base de datos correspondiente. Este servidor necesita tener instalados tanto *PHP* como *MySQL*, para su correcto funcionamiento.
- Por último, es necesario como mínimo un dispositivo con tecnología *iBeacon*. En general, la aplicación está hecha para que soporte cualquier modelo de baliza *BLE*. En concreto, los usados para este estudio son del fabricante *Estimote*.

A continuación, se muestran las especificaciones técnicas del hardware que se ha comentado anteriormente.

Ordenador de desarrollo	
Tipo	Portátil
Procesador	Intel Core i5 @ 1.60 GHz
Memoria RAM	2 GB
Disco duro	64 GB
Tarjeta gráfica	Intel HD Graphics 3000 288 MB
Sistema operativo	OS X Yosemite 64 bits
IDE Desarrollo	Xcode Versión 6.3.2

Tabla 51: Ordenador de desarrollo

Dispositivo IOS	
Tipo	Iphone
Procesador	Dual-core 1 GHz Cortex-A9



Memoria RAM	512 MB
Disco duro	16 GB
Tarjeta gráfica	PowerVR SGX543MP2
Sistema operativo	IOS 7.1

Tabla 52: Ordenador de pruebas

Dispositivo iBeacon	
Tipo	Estimote Beacon
Procesador	ARM Mo Cortex,
Duración batería	3 años; máximo 5años.
Datos modificables	Sí
Alcance máximo	> 70 metros

Tabla 53: Ordenador de pruebas

Los elementos requeridos para que un usuario pueda ejecutar la aplicación son:

- Por un lado, al igual que en el caso de desarrollo un dispositivo *IOS* con las mismas características, es decir, con *Bluetooth 4.0*.
- Por otro lado, un ordenador con acceso a internet y un navegador compatible con la aplicación web.
- Por último, tener acceso a algún dispositivo *iBeacon* para poder probar el correcto funcionamiento de la aplicación móvil o darlos de alta. Comentar que es necesaria una aplicación proporcionada por el fabricante de los *beacons*, para poder configurar los valores adecuados para cada estudio (*uuid, major, minor*).

Junto a estas especificaciones, se hace referencia al documento adjunto "*Anexo I: Guía rápida de configuración de un iBeacon*", que muestra al usuario los pasos a seguir para configurar el *beacon* con los valores adecuados para cada estudio. En este apartado, hay que destacar que la guía se realiza sobre la aplicación de un fabricante en concreto y, aunque los *beacons* a configurar utilizan la aplicación de otro fabricante, puede ser utilidad para ejemplificar como se configuran estos dispositivos.



## FASES DE IMPLEMENTACIÓN

A lo largo del desarrollo completo del presente proyecto, se realizarán una serie de tareas, dependientes entre sí, que llevarán al estado final del prototipo.

Se trata, pues, de las siguientes fases:

---

### APLICACIÓN MÓVIL

En este apartado se explicarán de forma separada las cuatro principales funcionalidades que la aplicación móvil necesita para su correcto funcionamiento.

---

#### DETECCIÓN ENTRA/SALIDA DE REGIÓN

Como se ha comentado en la sección “*Detección de dispositivos*” dentro del “*Estado del Arte*”, si queremos detectar un dispositivo teniendo la aplicación en segundo plano o parada, necesitamos los métodos que proporciona el *framework* de Apple (*CoreLocation*), que nos permitan detectar cuando se ha entrado en una región (rango de acción) de un *beacon*. En concreto estos métodos son, *didEnterRegion* y *didExitRegion*. Ambos métodos se ejecutan a través del método delegado *startMonitoringForRegion*, que permite el inicio de la monitorización en busca de *iBeacons*.

Cuando un usuario se instala la aplicación por primera vez, se ejecuta el método *startMonitoringForRegion* el cual indica al móvil que, cuando realice comprobaciones acerca de las conexiones y dispositivos de su entorno, sea avisado si se detecta algún *beacon*.

Una vez que el *smartphone* se lo notifica a la aplicación, se ejecutan los métodos de entrada o salida de región, según proceda. En el caso de que sea de salida, la aplicación espera un intervalo de tiempo hasta que, una vez realizadas varias comprobaciones, no se haya detectado ningún *beacon*. En el caso de entrada, la notificación se realiza nada más se comprueba que se ha detectado una baliza.



---

## OBTENER DATOS DEL IBEACON DETECTADO

Una vez se ha detectado la presencia de un *beacon*, el siguiente paso es la identificación de los valores *major* y *minor*. El problema de este apartado reside en que, el método que nos indica la entrada/salida de región no proporciona los valores del *beacon*, sólo su detección. Para obtener los esos valores necesitamos iniciar el método delegado *startRangingBeaconsInRegion*, el cual, una vez por cada escaneo que realice, ejecutará *didRangeBeacon:inRegion*. Este método se encarga de mostrarnos la información de la baliza. Sólo se ejecuta mientras la aplicación se encuentra activa, ni en segundo plano, ni parada.

Lo dicho en el párrafo anterior provoca que exista un problema a la hora de obtener los valores del *beacon* sin tener la aplicación activa. Este problema se pudo solucionar gracias a que cuando se detecta la entrada o salida de región, la aplicación se activa durante un breve periodo de tiempo, pudiendo aprovechar esos segundos para realizar un escaneo de los datos.

En el caso de que sean detectados varios *beacons* en un mismo escaneo (aunque en una situación ideal y bien configurada no debería pasar), la información que se guarda pertenece al primer dispositivo detectado, el resto se descartan. Esta situación hay que tenerla presente, porque se pueden producir entre 5 y 10 escaneos mientras la aplicación esta activa, por tanto, es lógico que una vez tengamos el primer *beacon* detectado, no se siga guardando datos.

En el caso de la salida de región, como al realizar el escaneo no se detecta ningún dato (porque se encuentra fuera de rango de ningún *beacon*), la solución adoptada consiste en guardar los valores proporcionados cuando se entra en una región.

---

## CONEXIÓN CON EL SERVICIO WEB

La conexión con la aplicación web se realiza una vez detectado el *beacon*, en el caso de disponer de conexión a internet. En caso contrario, se guarda toda la información obtenida y una vez que el usuario tenga conexión, se enviarán todos los datos al servidor.

La lógica que sigue el método de conexión para enviar los datos al servidor es la siguiente:

1. Guarda la fecha, la hora y el *Id* del dispositivo para ser enviados junto con el *major* del *beacon*.
2. Comprueba si dispone de conexión a la red. En caso negativo, se guardan los datos.
3. Si existe conexión a la red, se crea una petición *http POST*, a la dirección donde tenemos ubicado nuestro servidor escuchando, con los datos del dispositivo y el *beacon*. Los datos se transforman a *JSON*, para que puedan ser manipulados de manera sencilla por el servicio *php* al que va a ser enviado.



4. En caso de modificación del estado de la conexión de red a estado activa, se enviarán todos los datos guardados previamente.

---

## COMPROBACION CONEXIONES DE RED

Hay un par de consideraciones a tener en cuenta a la hora de comprobar la conexión a la red. Para comprobar el estado de red del dispositivo, necesitamos otro *framework* de *Apple* llamado *Reachability*. Cuando necesitemos comprobar si existe conexión o no, debemos crear un observador que informará del estado actual y de si se producen cambios. En el caso del uso de la aplicación sin conexión, hay que tener en cuenta una serie de limitaciones.

Por un lado, si se trabaja sin conexión puede que el dispositivo al estar suspendido y sin conexiones de red activas, no realice tantas comprobaciones de conectividad como las haría si tuviese *gps* o *wifi* activo. Por tanto, puede que la latencia en detectar la entrada o salida de región sea mucho mayor.

Por otro lado, en el caso de querer detectar si ha producido un cambio de conexión de red, debe iniciarse de nuevo la aplicación o tener la aplicación en segundo plano en el momento del cambio. En ningún caso la aplicación detectará el cambio si la aplicación está parada o el móvil suspendido.

---

## INTERFAZ DE USUARIO

La aplicación móvil, una vez instalada simplemente se debe cerrar o dejar en segundo plano y ya funcionará correctamente. Por tanto, en la interfaz de usuario sólo se muestra un mensaje con los detalles para su correcto funcionamiento, como se muestra en la siguiente imagen.

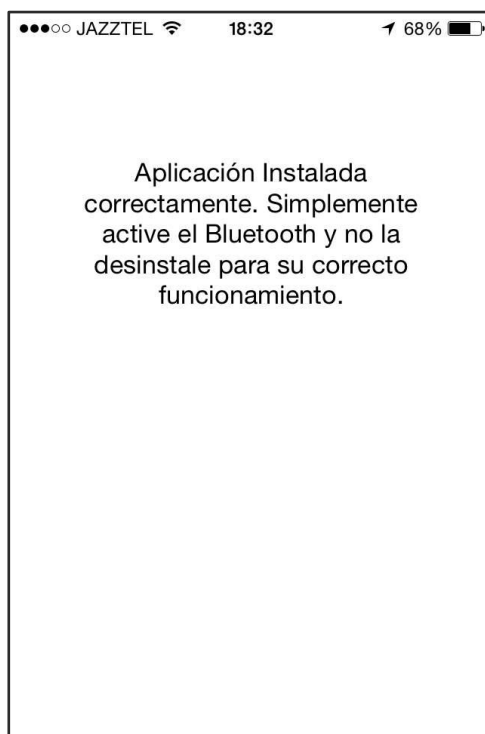


Ilustración 11: Pantalla inicio aplicación móvil

---

## APLICACIÓN WEB

La aplicación web se ha creado principalmente para dar soporte y mostrar los resultados obtenidos en la aplicación móvil. Aun así, se ha tratado de crear una aplicación coherente y funcional que permita el desarrollo de los casos de uso de manera eficiente.

En este apartado se explicarán de forma separada, las cinco principales funcionalidades que la aplicación web necesita para su correcto funcionamiento.

---

## INICIO DE SESIÓN Y REGISTRO DE USUARIOS

El soporte de roles y la creación de una sesión activa son partes relevantes de la creación de la aplicación web.



Se han creado dos roles claramente diferenciados que dan soporte a lo que se pretende con este prototipo. Por un lado, se ha creado el rol “usuario”, el cual puede visualizar y descargar estadísticas de los estudios. Por otro lado, se ha creado el rol “administrador” que, aparte de visualizar las estadísticas, tiene una serie de servicios para la gestión del proyecto. En concreto, tiene acceso a las secciones de “estudios”, “ibeacons” y “usuarios”.

El control de los roles se realiza mediante la página de autenticación y el control de la sesión activa de usuario. A continuación, se muestra la imagen de inicio de sesión.

Ilustración 12: Pantalla login aplicación web

En el caso de registrar una nueva persona en el sistema, por defecto su rol se asignará al valor “usuario”. En caso de necesitar permisos de administrador, es un usuario con el rol “administrador” el que modifique dicho permiso. Una vez autenticado el usuario como “administrador”, tendrá acceso a todos los servicios propios de su rol.

En cuanto a cuestiones de seguridad, las contraseñas están cifradas con el algoritmo *SHA256*, para evitar el guardado de las mismas en texto plano dentro de la base de datos. Se ha elegido este algoritmo porque es bastante seguro y da soporte a lo buscado en este proyecto.

El formulario para registrar un nuevo usuario se puede acceder desde en enlace “*Regístrate aquí*” de la sección “login”, mostrando la siguiente página:



Registrarse [Acceder](#)

**Nombre**

**Email**

**Contraseña**

[Registrarse](#)

Ilustración 13: Pantalla registro aplicación web

Como puede observarse, desde esta sección también podemos volver al apartado “login” con el enlace “Acceder”, situado en la esquina superior derecha.

---

## APARTADO IBEACONS Y ESTUDIOS

Estas secciones estarán disponibles siempre que el usuario inicie una sesión activa como administrador. En ambas secciones se ha creado un entorno que permite la creación, modificación y borrado de los *iBeacons* y estudios respectivamente.

En el caso de la sección de *iBeacons* se ha controlado, a la hora de dar de alta o modificar alguno de los *beacons* del sistema, que los datos a crear o modificar mantuviesen la coherencia junto con los datos del sistema alojados en la base de datos. Por ejemplo, no se puede modificar el valor del *major* una vez creado, porque es el identificador usado como clave primaria. Tampoco se permite añadir o modificar un *beacon* a un estudio que no esté dado de alta. A continuación, se muestra la interfaz de usuario donde se listan los *iBeacons* dados de alta en el sistema.





#### Administración de iBeacons

Nuevo iBeacon						
Mayor	Minor	X	Y	Etiqueta	IdEstudio	
48563	25136	1.2	3.7	Entrada	563	 
16572	50267	2.3	2.8	Juguetes	572	 

Ilustración 14: Pantalla iBeacons aplicación web

En la parte superior izquierda, se encuentra ubicado el botón para dar de alta a un *beacon*. En la parte derecha de cada fila de la tabla, se encuentran los botones para modificar o borrar cada *tupla*. Este esquema será idéntico para todos los apartados, salvo la sección de “estadísticas”.

En el caso de la sección de “estudios”, se ha controlado la presentación de las fechas en el formato adecuado, tanto en forma como en contenido. También se ha controlado el formato del identificador de estudio, siendo como máximo de tres cifras.

Las fechas indicadas en el estudio permiten controlar los datos insertados en la sección de “estadísticas”. A la hora de insertar un valor en la base de datos del sistema, si la fecha recibida se encuentra fuera de rango perteneciente a las “fechas inicial” y “final del estudio”, dicho valor no se insertará.

## APARTADO USUARIOS

Al igual que el apartado anterior, esta sección también estará disponible una vez iniciada una sesión como administrador. En esta sección se permite modificar y borrar un usuario, pero no se puede crear, ya que esa tarea corresponde a la sección de registro de usuarios.

En esta sección se ha controlado que el nombre de usuario no pueda ser modificado, porque es el identificador de dicho usuario, mientras que los campos de correo electrónico y rol si pueden. El correo electrónico deberá contener el formato adecuado y no estar relacionado con otro usuario. En el caso de la selección de rol, sólo se ha permitido la elección a los dos únicos roles disponibles.



## ESTADÍSTICAS

Esta sección está disponible desde cualquiera de los dos roles del sistema. Para poder visualizar datos de monitorización de un estudio o descargarse los datos, primero se debe seleccionar un estudio en concreto. Después, una vez visualizados podremos descargar tanto en los formatos *csv* o *xls* el contenido de los mismos.

A continuación, se muestra la disposición gráfica de la sección.

Datos de Monitorización estudio: 572

<div>Exportar a csv Exportar a Excel</div> <div>Selecciona Estudio</div>						
Id	Major	Minor	Fecha	Hora	Id Dispositivo	Entrada/Salida
13	16572	50267	17-06-2015	12:52:42	D3C86DFD-38FF-47C9-959C-7F10DB188706	1
12	16572	50267	17-06-2015	12:51:41	D3C86DFD-38FF-47C9-959C-7F10DB188706	0
14	16572	50267	19-06-2015	12:29:16	D3C86DFD-38FF-47C9-959C-7F10DB188706	1
17	16572	50267	19-06-2015	12:31:41	D3C86DFD-38FF-47C9-959C-7F10DB188706	0
18	16572	50267	20-06-2015	20:02:06	D3C86DFD-38FF-47C9-959C-7F10DB188706	0
19	16572	50267	20-06-2015	20:02:15	D3C86DFD-38FF-47C9-959C-7F10DB188706	1
20	16572	50267	20-06-2015	20:03:03	D3C86DFD-38FF-47C9-959C-7F10DB188706	0
21	16572	50267	20-06-2015	20:04:10	D3C86DFD-38FF-47C9-959C-7F10DB188706	1

Ilustración 15: Pantalla estadísticas aplicación web

En la esquina superior izquierda se encuentran los dos botones que permiten la descarga de los datos en los formatos *csv* y *Excel* y el desplegable en el que se puede seleccionar el estudio que se desea visualizar.

## ADAPTACIÓN INTERFAZ MÓVIL

El servicio web desarrollado es propenso a ser usado en dispositivos táctiles como *smartphones* o tabletas, por lo tanto necesita adecuarse a las resoluciones de pantalla de los mismos. Primero, se mostrará como aparece el menú que encabeza cada apartado, sin adaptarlo a una interfaz móvil.



Ilustración 16: Menú aplicación web

Como puede observarse, a la izquierda aparece el nombre del proyecto, las secciones "*Estadísticas*", "*iBeacons*", "*Estudios*" y "*Usuarios*", y a la derecha aparece el nombre de usuario y el botón de cierre de sesión.

Para la adaptación a entornos móviles se ha optado por un menú lateral en el que aparecerán todas las secciones una debajo de otra. Cuando el menú lateral se encuentra cerrado, la barra aparece de la siguiente manera:



Ilustración 17: Menú adaptación móvil aplicación web

En el caso desplegar el menú aparecerán mediante una animación las siguientes opciones:

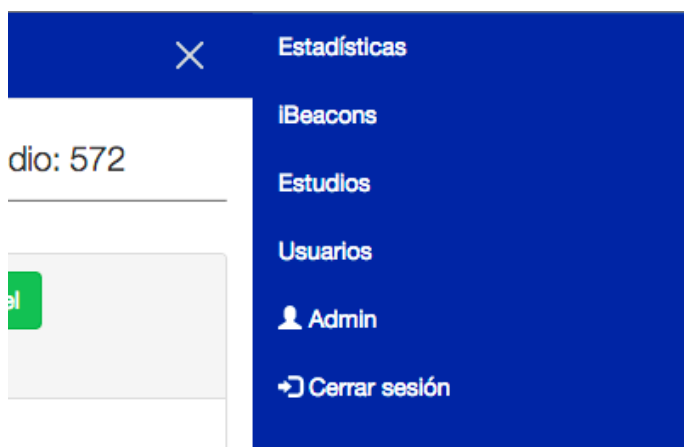


Ilustración 18: Menú adaptación móvil desplegado aplicación web



---

## CONEXIÓN BACK/FRONT END

Una vez explicado cómo funcionan ambas partes por separado, se pasa a comentar cómo se realiza la conexión entre ambas plataformas.

Por un lado, en la sección conexión al servicio web donde se comenta la funcionalidad de la aplicación móvil, se explica cómo se crea la petición desde la parte móvil. Una vez enviada esta petición al servidor se ha creado un servicio *php* que realiza las siguientes funciones

1. Primero comprueba que recibe una petición *POST*.
2. Después decodifica el archivo *JSON* y verifica que recibe el id del *beacon* (*major*) para comprobar que la petición *POST* se ha realizado correctamente.
3. Inicia una conexión con la base de datos para obtener las fechas de inicio y final del estudio asociado al *beacon* detectado.
4. Si la fecha enviada desde el *smartphone* se encuentra en el rango de las del estudio introduce los valores *major*, *fecha*, *hora*, *idDispositivo* y si es de entrada o salida en la base de datos.



## VALIDACIÓN Y EVALUACIÓN

Esta sección comprende un análisis de los requisitos estipulados con respecto a la dinámica de la aplicación.

### VALIDACIÓN DE REQUISITOS

Una vez completada la implementación de la aplicación, es preciso realizar un análisis del funcionamiento del sistema respecto a los requisitos establecidos, lo que refuerza las decisiones de diseño e implementación tomadas a lo largo del desarrollo del proyecto. Para ello, se toman en consideración todos los requisitos definidos en el apartado “*Requisitos del sistema*” del presente documento, y se verificará que se cumple dicho requisito a través del uso de la aplicación.

La tabla que se muestra a continuación sirve como plantilla para la validación de requisitos respecto a la implementación:

Requisito	
Descripción	
Prueba realizada	
Validación	

Tabla 54: Plantilla de validación de requisitos

A continuación se incluye una breve descripción de los campos que componen la tabla:

- **Identificador:** Este campo corresponde al identificador del requisito. Dicho identificador será único e inconfundible, de tal forma que cada requisito sea identificable sin posibilidad de error. Cada uno de los identificadores seguirá la siguiente nomenclatura:
  - **Requisitos funcionales:** RF-<XXX>-<YYY>, siendo <XXX>el valor app(referente a la aplicación móvil) o serv(referente al servicio web) y <YYY>un número entre 001 y 999.
  - **Requisitos no funcionales:** RNF-<XXX>-<YYY>, siendo <XXX>el valor app(referente a la aplicación móvil) o serv(referente al servicio web) y <YYY>un número entre 001 y 999.
- **Descripción:** Campo que incluye una descripción del requisito en cuestión.
- **Prueba:** Describe las acciones a realizar para comprobar el requisito.
- **Validación:** Determina si, con las pruebas realizadas, se puede afirmar que el requisito en cuestión se cumple durante la ejecución de la aplicación.

A continuación se muestran las tablas de validación para los requisitos de la aplicación.



Requisito	RF-app-001
Descripción	La aplicación detectará la entrada o salida del rango de los beacons dados de alta en el servidor.
Prueba	Se comprueba que al aproximarse a un beacon y al salirse de su radio de acción, el dispositivo es detectado.
Validación	Correcta

Tabla 55: Tabla de validación para RF-app-001

Requisito	RF-app-002
Descripción	Cada vez que el usuario entre o salga del rango de un beacon, la aplicación enviará al servidor información relativa al UID del dispositivo, al beacon asociado, fecha, hora y si entró o salió de la región (siempre que exista conexión de red).
Prueba	Se comprueba que al aproximarse a un beacon y al salirse de su radio de acción, la sección de estadísticas de la aplicación web refleja los nuevos datos.
Validación	Correcta

Tabla 56: Tabla de validación para RF-app-002

Requisito	RF-app-003
Descripción	Una vez instalada la aplicación en el terminal del cliente, la aplicación mostrará un mensaje en el que se informe que se ha instalado satisfactoriamente y las instrucciones para que funcione correctamente.
Prueba	Se comprueba que al iniciar la aplicación se muestra el mensaje.
Validación	Correcta

Tabla 57: Tabla de validación para RF-app-003

Requisito	RF-app-004
Descripción	La detección de entrada o salida de las regiones deberá realizarse en segundo plano, de la manera más transparentemente posible al usuario.
Prueba	Se comprueba que se detecta correctamente en las condiciones descritas, aunque en algunos casos puede existir cierta latencia es ser detectada la salida.
Validación	Correcta

Tabla 58: Tabla de validación para RF-app-004



Requisito	RF-app-005
Descripción	La aplicación detectará si el dispositivo tiene conexión a internet o no. En caso negativo, los datos se enviarán al servidor en diferido.
Prueba	Se comprueba que, en caso de tener conexión se envían los datos correctamente. En caso contrario, se guardan hasta que se detecte conexión y se envían.
Validación	Correcta

Tabla 59: Tabla de validación para RF-app-005

Requisito	RF-serv-001
Descripción	Tanto el rol administrador como el usuario tendrán acceso a la sección de estadísticas.
Prueba	Se comprueba que se puede acceder a la sección estadísticas con ambos roles.
Validación	Correcta

Tabla 60: Tabla de validación para RF-serv-001

Requisito	RF-serv-002
Descripción	El administrador tendrá acceso a la sección de gestión de iBeacons.
Prueba	Se comprueba que se puede acceder a la sección Ibeacons con el rol administrador.
Validación	Correcta

Tabla 61: Tabla de validación para RF-serv-002

Requisito	RF-serv-003
Descripción	El administrador tendrá acceso a la sección de gestión de estudios.
Prueba	Se comprueba que se puede acceder a la sección de estudios con el rol administrador.
Validación	Correcta

Tabla 62: Tabla de validación para RF-serv-003

Requisito	RF-serv-004
Descripción	El administrador tendrá acceso a la sección de gestión de usuarios.
Prueba	Se comprueba que se puede acceder a la sección de usuarios con el rol administrador.
Validación	Correcta

Tabla 63: Tabla de validación para RF-serv-004



Requisito	RF-serv-005
Descripción	La sección de estadísticas mostrará los valores relativos a la monitorización enviados por los dispositivos móviles.
Prueba	Se comprueba que el usuario puede visualizar los datos de monitorización enviados desde la aplicación móvil.
Validación	Correcta

Tabla 64: Tabla de validación para RF-serv-005

Requisito	RF-serv-006
Descripción	La sección de gestión de iBeacons mostrará una lista con todos los iBeacons existentes, pudiendo generar, editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).
Prueba	Se comprueba que, una vez iniciada sesión como administrador, se puede visualizar, crear, editar y borrar cualquier beacon del sistema.
Validación	Correcta

Tabla 65: Tabla de validación para RF-serv-006

Requisito	RF-serv-007
Descripción	La sección de gestión de estudios mostrará una lista con todos los estudios existentes, pudiendo generar, editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).
Prueba	Se comprueba que, una vez iniciada sesión como administrador, se puede visualizar, crear, editar y borrar cualquier estudio del sistema.
Validación	Correcta

Tabla 66: Tabla de validación para RF-serv-007

Requisito	RF-serv-008
Descripción	La sección de gestión de usuarios mostrará una lista con todos los usuarios existentes, pudiendo editar o eliminar cualquiera de ellos (siempre que esté autenticado como administrador).
Prueba	Se comprueba que, una vez iniciada sesión como administrador, se puede visualizar, editar y borrar cualquier usuario del sistema.
Validación	Correcta

Tabla 67: Tabla de validación para RF-serv-008

Requisito	RF-serv-009
Descripción	La opción de añadir iBeacon mostrará un formulario en la página que permitirá al usuario introducir valores para cada campo del





	iBeacon.
Prueba	Se comprueba que el formulario se muestra.
Validación	Correcta

Tabla 68: Tabla de validación para RF-serv-009

Requisito	RF-serv-010
Descripción	La opción de editar iBeacon mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho iBeacon.
Prueba	Se comprueba que el formulario se muestra.
Validación	Correcta

Tabla 69: Tabla de validación para RF-serv-010

Requisito	RF-serv-011
Descripción	La opción de eliminar iBeacon permitirá, previa confirmación del usuario, suprimir el iBeacon de la base de datos.
Prueba	Se comprueba que se permite esta opción.
Validación	Correcta

Tabla 70: Tabla de validación para RF-serv-011

Requisito	RF-serv-012
Descripción	La opción de añadir estudio mostrará un formulario en la página que permitirá al usuario introducir valores para cada campo del estudio.
Prueba	Se comprueba que el formulario se muestra.
Validación	Correcta

Tabla 71: Tabla de validación para RF-serv-012

Requisito	RF-serv-013
Descripción	La opción de editar estudio mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho estudio.
Prueba	Se comprueba que el formulario se muestra.
Validación	Correcta

Tabla 72: Tabla de validación para RF-serv-013

Requisito	RF-serv-014
Descripción	La opción de eliminar estudio permitirá, previa confirmación del



	usuario, suprimir el estudio de la base de datos.
Prueba	Se comprueba que se permite esta opción.
Validación	Correcta

Tabla 73: Tabla de validación para RF-serv-014

Requisito	RF-serv-015
Descripción	La opción de editar usuario mostrará un formulario en la página que permitirá modificar, campo a campo, los valores introducidos previamente para dicho usuario.
Prueba	Se comprueba que el formulario se muestra.
Validación	Correcta

Tabla 74: Tabla de validación para RF-serv-015

Requisito	RF-serv-016
Descripción	La opción de eliminar usuario permitirá, previa confirmación del usuario, suprimir el usuario de la base de datos.
Prueba	Se comprueba que se permite esta opción.
Validación	Correcta

Tabla 75: Tabla de validación para RF-serv-016

Requisito	RF-serv-017
Descripción	La interfaz de usuario deberá adaptarse a dispositivos móviles.
Prueba	Se comprueba que si se reduce el tamaño de la pantalla del navegador, la interfaz se adapta al nuevo tamaño.
Validación	Correcta

Tabla 76: Tabla de validación para RF-serv-017

Requisito	RF-serv-018
Descripción	Las contraseñas de los usuarios del servicio serán encriptadas con el algoritmo SHA256.
Prueba	Se comprueba que se guarda el valor encriptado de las contraseñas en la base de datos.
Validación	Correcta

Tabla 77: Tabla de validación para RF-serv-018

Requisito	RF-serv-019
Descripción	El sistema podrá descargar los resultados obtenidos de los estudios



	en los formatos Excel y csv.
Prueba	Se comprueba que el sistema permite la descarga de los datos en ambos formatos.
Validación	Correcta

Tabla 78: Tabla de validación para RF-serv-019

Requisito	RNF-app-001
Descripción	La aplicación podrá ejecutarse en el sistema operativo IOS versión 7.1.
Prueba	Se comprueba que la aplicación funciona correctamente en la versión de sistema operativo especificada.
Validación	Correcta

Tabla 79: Tabla de validación para RNF-app-001

Requisito	RNF-app-002
Descripción	La aplicación funcionará con Bluetooth 4.0.
Prueba	Se comprueba que la aplicación funciona correctamente usando la tecnología Bluetooth 4.0.
Validación	Correcta

Tabla 80: Tabla de validación para RNF-app-002

Requisito	RNF-app-003
Descripción	La aplicación será capaz de mantener un rendimiento óptimo en dispositivos de diferentes características.
Prueba	Al disponer de sólo un dispositivo, no se ha podido probar este requisito.
Validación	No verificado

Tabla 81: Tabla de validación para RNF-app-003

Requisito	RNF-app-004
Descripción	La aplicación requerirá un mínimo de 2 megas de espacio en disco duro para su instalación.
Prueba	Se comprueba que la aplicación se instala correctamente si dispone del espacio libre indicado.
Validación	Correcta

Tabla 82: Tabla de validación para RNF-app-004



Requisito	RNF-serv-001
Descripción	El servicio web se podrá ejecutar desde cualquiera de los siguientes navegadores: Chrome, Firefox, Internet Explorer y Safari.
Prueba	Se comprueba que se visualiza correctamente en cualquiera de los 4 navegadores mencionados.
Validación	Correcta

Tabla 83: Tabla de validación para RNF-serv-001

Requisito	RNF-serv-002
Descripción	Los datos del servicio web sólo podrán ser editados por los usuarios autenticados que tengan permiso para ello.
Prueba	Se comprueba que los datos del servicio web sólo pueden ser editados por usuarios autenticados.
Validación	Correcta

Tabla 84: Tabla de validación para RNF-serv-002

Según se aprecia en las tablas anteriores, todos los requisitos se cumplen correctamente salvo en un caso.

Aunque el requisito no se haya podido verificar, con mucha probabilidad la aplicación mantendrá un rendimiento correcto, ya que no necesita de muchos recursos para su correcto funcionamiento.

Como conclusión para este análisis se puede afirmar que la propuesta implementada sigue los requisitos establecidos al comienzo del desarrollo.



## GESTIÓN DEL PROYECTO

En este apartado se exponen las planificaciones estimada y final para el proyecto, junto al presupuesto estimado para el mismo.

### PLANIFICACIÓN INICIAL

Al comienzo de este proyecto, el 1 de Febrero, se estimó una planificación del tiempo que se tardaría en realizar las múltiples y diferentes tareas de las que consta. Debido a que durante todo el proyecto de lunes a viernes trabajaba en una empresa, se fijó un cálculo aproximado de 6 horas los sábados y otras 6 los domingos, durante los meses de febrero, marzo, abril, mayo y junio. Se calculó que el trabajo estaría terminado para mitad de junio.

El trabajo se ha dividido en varias fases perfectamente diferenciadas. En primer lugar se realiza la fase de estudio del problema, junto a un análisis y aprendizaje de las tecnologías a utilizar durante el desarrollo. A continuación se realiza la fase de implementación, durante y al final de la cual se harían unas pruebas de la aplicación. Por último se procede a desarrollar la documentación del trabajo realizado.

La fase de estudio, análisis y aprendizaje tiene varios objetivos principales: analizar el problema junto a trabajos realizados y alternativas del mercado para dar con la mejor propuesta o solución, investigar sobre las tecnologías apropiadas para la propuesta y, una vez establecidas las tecnologías, realizar un aprendizaje de las mismas para conocer el entorno en el que se desarrollará la solución propuesta.

Durante la fase de implementación se desarrolla la aplicación móvil en primer lugar. Una vez comprobado que funciona correctamente, se pasó a desarrollar la parte web. Por último, se implementó la conexión que comunica ambas plataformas.

La primera fase de pruebas se realiza prácticamente de modo simultáneo a la de implementación, tanto en el caso de la aplicación móvil como de la aplicación web, pues es necesario comprobar que los algoritmos y los ajustes que se van desarrollando funcionan apropiadamente.

La segunda fase de pruebas se realiza al final de la implementación. En base a los resultados se realizan pequeños ajustes y retoques sobre todo en el apartado de la aplicación web y comunicación. Como resultado de las pruebas, la validación de requisitos resultó satisfactoria.

Por último se escribirá la documentación del proyecto. El resultado será un documento en el que se describirá el trabajo realizado a lo largo del desarrollo del proyecto, evaluando los resultados obtenidos.



## PLANIFICACIÓN FINAL

El diagrama de Gantt muestra la planificación final de tiempo invertido durante el desarrollo de este trabajo de fin de grado.

Debido al gran tamaño del gráfico, se incluye como documento adjunto en el apartado "*Anexo II: Diagrama de planificación*".

El desarrollo del proyecto ha transcurrido por lo general según lo planificado, salvo por una serie de diferencias. En primer lugar, los meses de inicio del proyecto coincidieron con épocas de trabajo intensivas, por lo que el rendimiento de los primeros fines de semana fue menor.

El parón que aparece en la semana del 23 al 29 de marzo se debe a una salida vacacional del alumno, continuando a su vuelta el desarrollo de la parte web.

Por último, algunos cambios y retoques de última hora retrasaron un poco más de lo debido la finalización del proyecto.

Pese a estos inconvenientes se considera que la planificación inicial fue correcta, ya que la planificación real se aproxima a la estimada.



## PRESUPUESTO

Este apartado tiene como objetivo el cálculo de un presupuesto estimado para el desarrollo del presente proyecto.

La versión final de la aplicación puede llegar a tener algún cierto valor económico, por lo que resulta necesario crear un presupuesto que permita conocer los gastos asociados al personal y al material amortizable del que se dispone durante el desarrollo.

El coste de los recursos humanos para el proyecto incluye al alumno y al tutor del trabajo, considerando que el número de horas trabajadas por el tutor es un cuarto de la correspondiente a la del alumno (ofrece ayuda y asistencia durante el proyecto pero tienen sus propias labores que desarrollar en sus puestos de trabajo).

Se tomará el gasto de personal de la siguiente forma:

- El coste estimado para el alumno, considerado como programador con baja experiencia a efectos prácticos, será de 6.25€/hora (12.000 € brutos/año) [22].
- El coste estimado para el tutor, considerado como jefe de proyecto con alta experiencia a efectos prácticos, será de 15.65 €/hora (30.000 € brutos/año) [22].

El gasto de personal queda, pues, de la siguiente forma:

Gastos alumno		
Fase del desarrollo	Horas trabajadas (h)	Coste (€)
Estudio, análisis y aprendizaje	40	250
Implementación y pruebas	170	1062.5
Documentación	40	250
Total	250	1562.2
Gastos tutor		
Fase del desarrollo	Horas trabajadas (h)	Coste (€)
Estudio, análisis y aprendizaje	10.75	168.23
Implementación y pruebas	9.25	144.76
Documentación	11	172.15
Total	31	485.14
Total del personal	281	2047.3

Tabla 85: Gastos de personal

En la siguiente tabla se muestran los gastos relativos al material amortizable, en los que se encuentran los productos hardware y software utilizados en el proyecto.



Tipo	Artículo	Precio (€)	Amortización (meses)	Amortización (€/mes)	Periodo de uso (meses)	Coste en proyecto (€)
Hardware	Ordenador portátil	880	60	14.70	5	73.50
	Dispositivo Iphone	699.00	60	11.65	5	58.25
	2 iBeacons	60	60	1	5	5
Software	Licencia Desarrollador Apple	100.00	48	2.08	5	10.41
Total						147.16

Tabla 86: Gastos de material amortizable

Los gastos directos, compuestos por los totales de las tablas anteriores, son 2194.50 €, mientras que los gastos indirectos, como gastos de luz y agua, se estiman en torno al 5% de los gastos directos. Por otra parte, en caso de que el trabajo se vendiese, el margen de beneficios previsto sería del 5% de los gastos. El presupuesto total queda de la siguiente forma:

Concepto	Coste (€)
Gastos directos	2194.50
Gastos indirectos	152.16
Total del presupuesto	2346.66

Tabla 87: Presupuesto total del proyecto





## CONCLUSIONES

Este apartado constituye las conclusiones finales sobre el desarrollo del presente trabajo de fin de grado. Asimismo se incluyen unas líneas finales sobre las diferentes mejoras que pueden aplicarse sobre la aplicación para favorecer una mayor experiencia de usuario.

## COMENTARIOS FINALES

El objetivo que se planteó para el proyecto al comienzo de su desarrollo fue la creación de una aplicación que permitiese la recolección de datos de usuario, monitorizando todo el comportamiento del cliente. Aparte, se pretendía realizar esta monitorización de la forma más transparente al usuario y en tiempo real. Tras concluir el desarrollo de la aplicación y realizar una serie de pruebas, se puede decir que, pese a no ofrecer un tiempo completamente real a la hora de salir de las regiones (debido a su latencia), los datos reflejados en el servidor informan correctamente de los comportamientos que un cliente puede realizar, mientras simplemente lleva su dispositivo en el bolsillo y la aplicación instalada. Todo esto que permite afirmar que se ha conseguido cumplir el objetivo principal y los secundarios que se establecieron en un principio.

## LÍNEAS FUTURAS

Dado que es una tecnología muy novedosa, posiblemente en un futuro se mejoren muchos de los aspectos actuales de los dispositivos *iBeacons*.

Respecto al proyecto en sí, existen algunas posibles mejoras a tener en cuenta. En concreto, a la hora de la visualización de estadísticas, se podrían mostrar ciertas gráficas que mejorasen la interpretación de los datos. Por ejemplo, gráficos de barras con los *beacons* más detectados, horas del día, lugares más visitados, rutas más frecuentes, etc...

Por otro lado, se podría mostrar un mapa con los puntos en los que se encuentran los *beacons* y cómo los usuarios van recorriendo el establecimiento en tiempo real.

En el caso de las cuestiones relativas a la seguridad en los *beacons*, habría que estudiar la posibilidad de utilizar *beacons* de distintos fabricantes para utilizar el *framework* más seguro posible. Este aspecto



también es claramente mejorable con el paso del tiempo, ya que existen fabricantes que han identificado estos problemas y están tratando de resolverlos.

El objetivo principal de este proyecto consistió en proporcionar todos los datos posibles mediante la monitorización. La siguiente fase sería el análisis e interpretación de los mismos para crear patrones de conducta, identificar usuarios, rutas, como influyen ciertos factores como fechas, horas concretas, etc...

Toda esta segunda fase tendría un componente claro de estudio de tecnologías relacionadas con el *big data* y minería de datos.



## REFERENCIAS

- [1] J.M.S, "Bluetooth: la historia de un sistema basado en un rey danés", Abc, consulta: 5 junio 2015, <http://www.abc.es/tecnologia/moviles/20141021/abci-bluetooth-historia-futuro-201410201659.html>
- [2] Bluetooth SIG, consulta: 27 mayo 2015, <http://locationbasedserviceevent.com/company/bluetooth-sig/>
- [3] Jim Kardach, "Tech History: How Bluetooth got its name.", Eetimes, consulta: 12 mayo 2015, [http://www.eetimes.com/document.asp?doc\\_id=1269737](http://www.eetimes.com/document.asp?doc_id=1269737)
- [4] Mundo LG, "¿Cómo funciona el bluetooth?", consulta: 27 mayo 2015, <http://www.lgblog.cl/2015/04/01/como-funciona-la-tecnologia-bluetooth/>
- [5] Rolf Nilsson and Bill Saltzstein, "Bluetooth Low Energy vs. Classic Bluetooth: Choose the Best Wireless Technology For Your Application", consulta: 4 junio 2015, <http://www.medicalelectronicsdesign.com/article/bluetooth-low-energy-vs-classic-bluetooth-choose-best-wireless-technology-your-application>
- [6] C2 Digital Agency, "¿Qué son los iBeacons y para qué sirven?", consulta: 13 junio 2015, <http://blog.c2digitalagency.es/que-son-los-ibeacons-y-para-que-sirven>
- [7] Anónimo, "Estimote's new Platform, even as Beacons Tech gets overshadowed by NFC", consulta: 30 mayo 2015, <http://letstalkpayments.com/estimotes-new-platform-even-beacons-tech-gets-overshadowed-nfc/>
- [8] iBeacons for Developers, Apple, consulta: 17 mayo 2015, <https://developer.apple.com/ibeacon/>
- [9] Iván González, "Bluetooth Low Energy vs Wi-Fi", InnoQuant, consulta: 15 mayo 2015, <http://blog.innoquant.com/2014/05/bluetooth-low-energy-vs-wi-fi/>
- [10] Matthew Szymczyk, "¿Beacon vs. NFC Infographic?", Zugará, consulta: 24 mayo 2015, <http://zugará.com/beacon-vs-nfc-infographic>
- [11] Activa Internet, "NFC, ¿qué es y para qué sirve?", consulta: 17 mayo 2015, <http://www.activainternet.es/nfc-tecnologia-pymes/>
- [12] RFID, Flatorb, Marzo 2015, consulta: 8 junio 2015, <http://www.flatorb.com/rfid/>
- [13] Joe Keating, Infinite Power Solutions, "BLE and energy harvesting enable low-cost, maintenance-free active RFID systems", consulta: 29 mayo 2015, <http://www.ecnmag.com/articles/2012/05/ble-and-energy-harvesting-enable-low-cost-maintenance-free-active-rfid-systems>



- [14] Developer Documentation, Estimote, consulta: 8 junio 2015, <http://developer.estimote.com/documentation/>
- [15] Choosa, "Estimote inaugura el Internet de las Cosas", consulta: 11 mayo 2015, [http://www.choosa.net/pt/p/Choosa/Estimote inaugura el Internet de las Cosas](http://www.choosa.net/pt/p/Choosa/Estimote%20inaugura%20el%20Internet%20de%20las%20Cosas)
- [16] Miguel López, "La tecnología iBeacon de Apple logra aumentar las ventas de los McDonalds en un estudio", Applesfera, consulta: 5 junio 2015, <http://www.applesfera.com/iphone/la-tecnologia-ibeacon-de-apple-logra-aumentar-las-ventas-de-los-mcdonalds-en-un-estudio>
- [17] Kote Puerto, "JIVR es una bici plegable y eléctrica, que además soporta iBeacon", Xataka, consulta: 20 mayo 2015, <http://www.xataka.com/vehiculos/jivr-es-una-bici-plegable-y-electrica-que-ademas-soporta-ibeacon>
- [18] David Ramos, "Aena implanta tecnología 'iBeacons' para informar a los pasajeros", Europa Press, consulta: 4 junio 2015, <http://www.europapress.es/turismo/transportes/aeropuertos/noticia-aena-implanta-tecnologia-ibeacons-informar-pasajeros-20150428124343.html>
- [19] Jesús Pacheco Búrdalo, "La MLB introducirá iBeacon de Apple en 20 estadios de béisbol en marzo", consulta: 9 junio 2015, <http://www.gadgetos.com/noticias/mlb-introducira-ibeacon-apple-20-estadios-beisbol-marzo/>
- [20] Pablo, "Presentado el App Oficial del Quijote 2015 en Fitur. Tecnología iBeacon en turismo", consulta: 26 mayo 2015, <http://www.securware.com/en/presentado-el-app-oficial-del-quijote-2015-en-fitur-tecnologia-ibeacon-en-turismo/>
- [21] Eugenio Cabezas, "Rincón de la Victoria recibe 371.000 euros para implantar el proyecto 'Smart Costa del Sol'", Diario Sur, consulta: 14 mayo 2015, <http://www.diariosur.es/axarquia/20150312/rincon-victoria-recibe-euros-20150312221412.html>
- [22] Tendencia de salario para ingenieros informáticos, Infojobs Trends, consulta: 7 junio 2015, <http://plandecarrera.infojobs.net/puesto-de-trabajo/ingeniero-informatico>

## ANEXO I: GUÍA RÁPIDA DE CONFIGURACIÓN DE UN IBEACON

En este documento de referencia se pretende especificar los pasos necesarios que debe seguir el usuario de la aplicación para configurar los *beacons* que desee asociar a un estudio.

En concreto, la guía se realizará para la configuración de los *beacons* del fabricante *Estimote*. Cabe destacar, que en caso de tener unos dispositivos de diferente fabricante, esta guía también puede ser de utilidad, aunque no se sigan los mismos pasos puede ayudar a entender el concepto de modificación de las balizas.

Para empezar debemos descargarnos la aplicación disponible en la *AppStore*, en este caso la aplicación se llama *Estimote*.

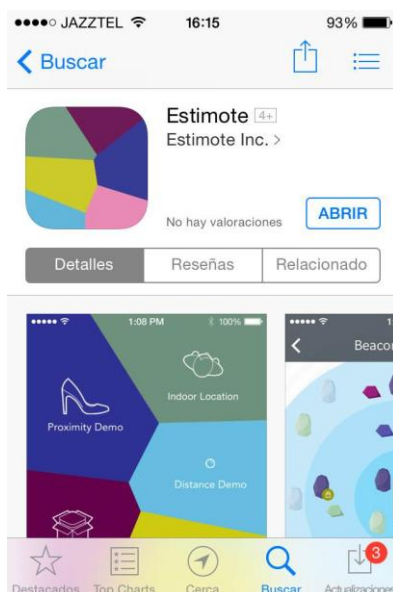


Ilustración 19: Aplicación Estimote en App Store.

Una vez descargada, para continuar debemos tener activo la opción “*bluetooth*” del *smartphone* y estar en rango de un *beacon* que emita señales.

A continuación, iniciamos la aplicación y seleccionamos en su menú principal la opción “*Devices*”.



Ilustración 20: Menú principal aplicación Estimote

En la siguiente pantalla se mostrará una especie de radar indicando los dispositivos detectados por la aplicación, así como su proximidad.

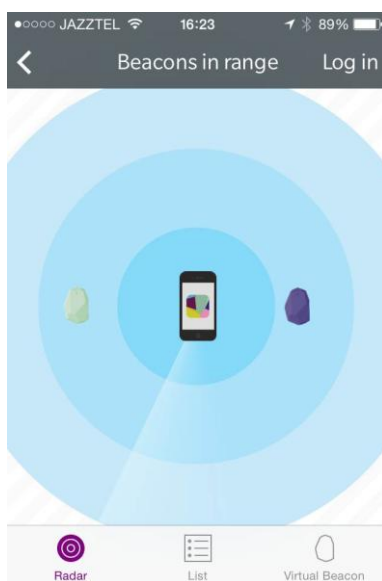


Ilustración 21: Sección radar aplicación Estimote



En la parte inferior aparecerán tres opciones: "Radar", "List" y "Virtual Beacon". La opción "Radar" es la que se visualiza al iniciar la pantalla por defecto ya comentada. La opción "Virtual Beacon" permite convertir nuestro dispositivo como un emisor *beacon* virtual. Por último, la opción "List" nos muestra un listado con los *beacons* detectados.

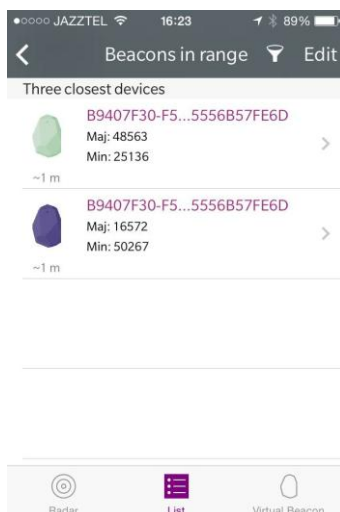


Ilustración 22: Sección "List" aplicación Estimote

Si seleccionamos uno de los ítems de la lista, podremos modificar algunos parámetros del *beacon* seleccionado, siempre que seamos dueños de los dispositivos (pide autenticación para el siguiente paso). Una vez autenticado como dueño del *beacon* a modificar, aparecerá una lista con los parámetros modificables. Por ejemplo, la intensidad o frecuencia de la señal, actualizar el *firmware*, etc...

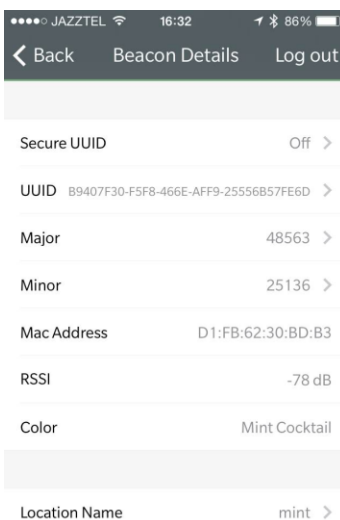


Ilustración 23: Propiedades de un beacon detectado



En nuestro caso, como sólo necesitamos modificar los valores *mayor* y *minor*, los seleccionamos e indicamos los nuevos valores.

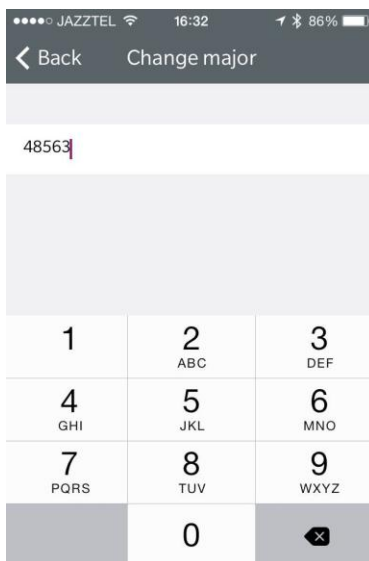


Ilustración 24: Modificación Major aplicación Estimote

Una vez modificados, ya tendríamos los valores correctos para nuestro estudio.





## ANEXO II: DIAGRAMA DE PLANIFICACIÓN

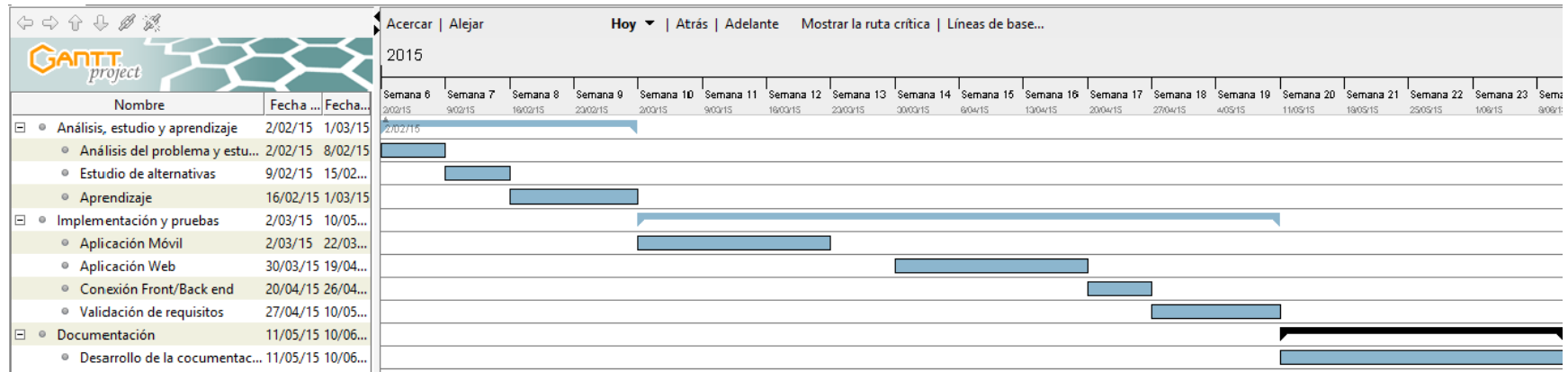


Ilustración 25: Diagrama de planificación inicial estimada